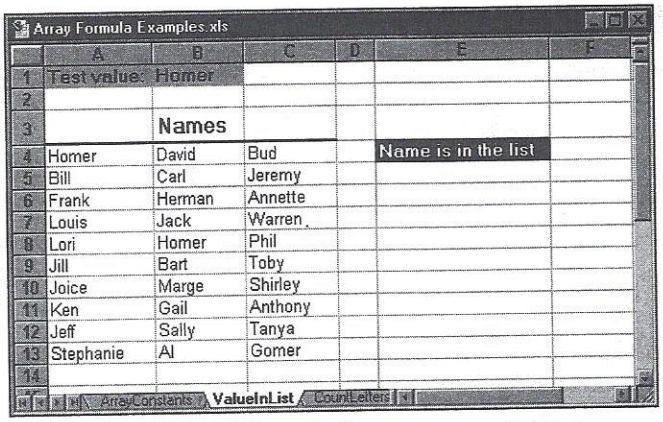


Figure 20-9: Determining whether a range contains a particular value.

The formula in cell E4 is as follows:

```
{=IF(OR(TestValue=Names),"Name is in the list","Name not
found")}
```

This formula compares TestValue to each cell in the range Names. It builds a new array that consists of logical TRUE or FALSE values. The OR function returns TRUE if any one of the values in the new array is TRUE. The IF function determines which message to display based on the result.

Counting Characters in a Range

This example demonstrates how to use nested functions in an array formula to loop through each element in the input range. Figure 20-10 shows a worksheet with text entered in a range named WordList.

The array formula in cell B1 is as follows:

```
{=SUM(LEN(WordList))}
```

This formula is quite straightforward. It creates an array that consists of the length of each word in the WordList range. Then, it uses the SUM formula to add the values in this new array. You can accomplish this without an array formula by using an additional column of formulas and then summing the results.

Array Formula Examples.xls

	A	B	C	D	E	F
1	**Wordlist**	74	= Number of letters in WordList			
2	January					
3	February					
4	March					
5	April					
6	May					
7	June					
8	July					
9	August					
10	September					
11	October					
12	November					
13	December					
14						

ValueInList | CountLetters | MaxMin_Chang

Figure 20-10: This array formula counts the number of characters in a range of text.

Computing Maximum and Minimum Changes

Figure 20-11 shows another example of how an array formula can eliminate the need for intermediary formulas. This worksheet shows two test scores for a group of students. Array formulas compare the two tests and calculate the largest decrease and the largest increase.

Array Formula Examples.xls

	A	B	C	D	E	F
1						
2		**Test 1**	**Test 2**			
3	Student 1	45	56		-5	Largest Decrease
4	Student 2	78	73		11	Largest Increase
5	Student 3	91	93			
6	Student 4	62	69			
7	Student 5	74	71			
8	Student 6	87	87			
9	Student 7	81	89			
10	Student 8	83	80			
11	Student 9	53	53			
12						

CountLetters | MaxMin_Change | Su

Figure 20-11: Array formulas determine the largest decrease and the largest increase in test scores.

The formulas are as follows:

```
E3:      {=MIN(C3:C11-B3:B11)}
E4:      {=MAX(C3:C11-B3:B11)}
```

Looping Through Characters in a Cell

The following array formula calculates the sum of the digits in an integer, which is stored in a cell named Number:

```
{=SUM(VALUE(MID(Number,ROW($A$1:OFFSET($A$1,LEN(Number)-
1,0)),1)))}
```

This is a rather complex formula that makes use of an interesting trick; thus, it is explained next one part at a time, so that you can see how it works. (Figure 20-12 shows an example.)

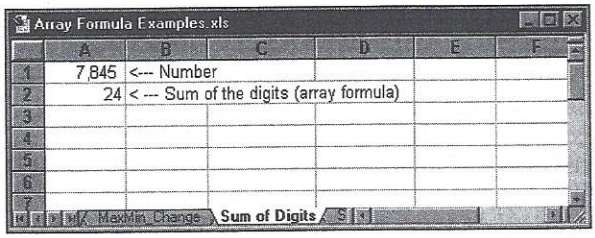


Figure 20-12: An array formula calculates the sum of the digits in a value.

You may be confused by the ROW function (this is the trick). This function is used to generate an array of consecutive integers, beginning with 1 and ending with the number of digits in the absolute value of Number.

If Number is 489, then LEN(Number) is 3. The ROW function can then be simplified as follows:

```
{=ROW($A$1:OFFSET($A$1,3-1,0))}
```

This formula generates an array with three elements: {1,2,3}, which is used as the second argument for the MID function (the third argument is 1). The MID part of the formula, simplified a bit and expressed as values, is the following:

```
{=MID(489,{1,2,3},1)}
```

This formula generates an array with three elements: {4,8,9}. By simplifying again and adding the SUM function, the formula becomes as follows:

```
{=SUM({4,8,9})}
```

This produces the result of 21.

The following is another version of this formula that also works with negative numbers. The ABS function is added to calculate the absolute value of the result:

```
{=SUM(VALUE(MID(ABS(Number),ROW($A$1:OFFSET($A$1,LEN(ABS(Number
))-1,0)),1)))}
```

Summing Every *n*th Value in a Range

The next example can be quite useful. Suppose that you have a range of values and you want to compute the sum of every third value in the list — the first, the fourth, the seventh, and so on. You can't accomplish this with a standard formula. The following array formula does the job, however. It assumes that a cell named Nth determines which values to sum, and that the range to sum is named Data.

```
{=IF(nth=0,0,SUM(IF(MOD(ROW($A$1:OFFSET($A$1,COUNT(Data)-
1,0)),nth)=0,Data,0)))}
```

The formula uses the MOD function to determine which values to sum. The first argument for the MOD function is as follows:

```
ROW($A$1:OFFSET($A$1,COUNT(Data)-1,0))
```

This expression generates an array that begins with 1 and ends with the number of cells in the Data range. If the MOD function returns 0, the value is included in the array to sum.

Notice that a special case exists for when Nth is 0 (that is, sum every cell in the range), because the MOD function returns an error when its second argument is 0.

This formula has a limitation: It works only when Data consists of a single column of values, because it uses the ROW function to determine the element in the array.

Figure 20-13 shows an example that uses the preceding array formula, plus a series of intermediary formulas to calculate the result without using an array formula.

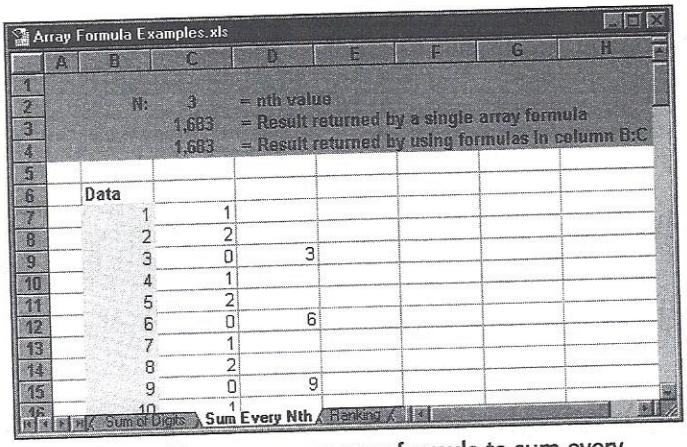


Figure 20-13: You can use an array formula to sum every *n*th element in a range—or use a series of intermediary formulas (a less-efficient approach).

An Alternate Method of Ranking

Often, computing rank orders for a range of data is helpful. If you have a worksheet with the annual sales figures for 20 salespeople, for example, you may want to know how each person ranks, from highest to lowest.

If you do this sort of work, you've probably discovered Excel's RANK function. You also may have noticed, however, that the ranks produced by this function don't handle ties the way that you may like. For example, if two values are tied for third place, they both receive a rank of 3. Many people prefer to assign each an average (or midpoint) of the ranks—that is, a rank of 3.5 for both values tied for third place.

Figure 20-14 shows a worksheet that uses two methods to rank a column of values (named Sales). The first method (column C) uses Excel's RANK function. Column D uses array formulas to compute the ranks.

The following is the array formula in cell D2:

```
{=IF((SUM(IF(Sales=B2,1)))=1,(SUM(IF(Sales>=B2,1,0))),(SUM(IF(S
ales>=B2,1)))-((SUM(IF(Sales=B2,1)))-1)*0.5)}
```

This formula was entered into cell D2 and then copied to the cells below it.

The formula is rather complex, but breaking it down into parts should help you understand how it works.

Array Formula Examples.xls

	A	B	C	D	E	F	G	H
1	Salesperson	Sales	Excel's Rank Function	Ranks With Array Formula				
2	Adams	123,000	6	6				
3	Bigelow	98,000	9	10				
4	Fredericks	98,000	9	10		Assigned middle rank		
5	Georgio	98,000	9	10				
6	Jensen	25,000	12	12				
7	Juarez	101,000	8	8				
8	Klein	305,000	1	1				
9	Lynch	145,000	3	3.5				
10	Mayne	145,000	3	3.5		Assigned average rank		
11	Roberton	121,000	7	7				
12	Slokum	124,000	5	5				
13	Wu	150,000	2	2				
14								

Sum of Digits | Sum Every Nth | Ranking | Discrete

Figure 20-14: Ranking data with Excel's RANK function and with array formulas.

Frequency Distributions

Before Excel 5, the only way to calculate frequency distributions was to use array formulas. Beginning with Excel 5, however, the COUNTIF function provides a more direct way to generate frequency distributions.

Figure 20-15 shows a worksheet with a series of scores in column A that range from 1 to 4. Column D contains array formulas to calculate the frequency of each score.

Array Formula Examples.xls

	A	B	C	D	E	F	G
1							
2	Scores		Score	Array Formulas	COUNTIF Formulas		
3	1		1	6	6		
4	3		2	7	7		
5	2		3	5	5		
6	4		4	6	6		
7	4						
8	2						
9	3						
10	3						
11	2						
12	1						
13	1						

Sum Every Nth | Ranking | Discrete

Figure 20-15: Calculating discrete frequency distributions by using array formulas and COUNTIF functions.

The formula in D6 is as follows:

```
{=SUM(IF(Scores=C3,1))}
```

The corresponding formulas in column E use the COUNTIF function. The following is the formula in cell E6:

```
=COUNTIF(Scores,C3)
```

Both of these methods count specific values. But what if the scores are noninteger values, as in Figure 20-16? Both types of formulas require modification to handle noninteger data.

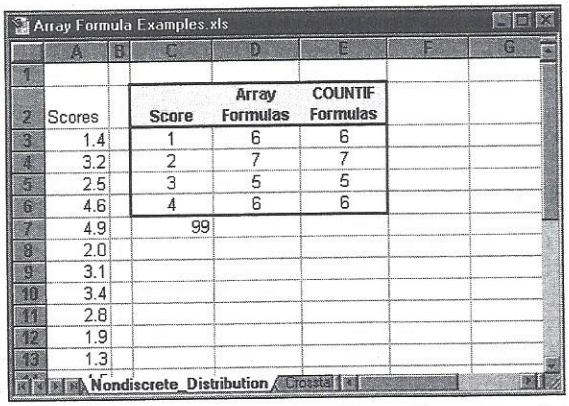


Figure 20-16: Calculating nondiscrete frequency distributions by using array formulas and COUNTIF functions.

The array formula can be modified as follows:

```
=SUM(IF(Scores>=C3,1))-SUM(IF(Scores>=C4,1))
```

The following is the revised COUNTIF formula:

```
=COUNTIF(Scores,">="&C3)-COUNTIF(Scores,">="&C4)
```

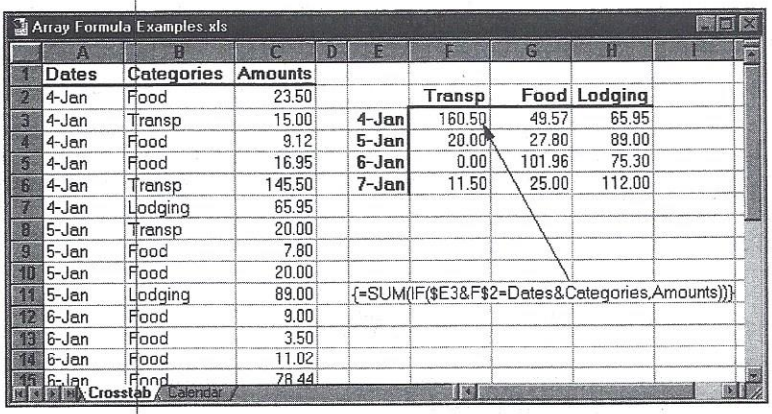
The array formula requires you to add an additional value in column C, so that the last array formula doesn't refer to an empty cell (I added a value of 99).

You also can compute distributions by using the Histogram tool in the Analysis ToolPak (see Chapter 28). An advantage to using arrays or COUNTIF functions, however, is that these procedures are dynamic and display the correct values if you change the input data.

Dynamic Crosstabs

The preceding section demonstrates that using COUNTIF is better than using array formulas to calculate frequency distributions. This section demonstrates how to extend these distributions into another dimension and create crosstabs. In this case, an array formula is the only method that can get the job done. This technique enables you to create a dynamic crosstab table that is updated automatically whenever the data is changed. Even a pivot table can't do that!

The worksheet in Figure 20-17 shows a simple expense account listing. Each item consists of the date, the expense category, and the amount spent. Each column of data is a named range, indicated in the first row.



	A	B	C	D	E	F	G	H	I
1	Dates	Categories	Amounts						
2	4-Jan	Food	23.50			Transp	Food	Lodging	
3	4-Jan	Transp	15.00		4-Jan	160.50	49.57	65.95	
4	4-Jan	Food	9.12		5-Jan	20.00	27.80	89.00	
5	4-Jan	Food	16.95		6-Jan	0.00	101.96	75.30	
6	4-Jan	Transp	145.50		7-Jan	11.50	25.00	112.00	
7	4-Jan	Lodging	65.95						
8	5-Jan	Transp	20.00						
9	5-Jan	Food	7.80						
10	5-Jan	Food	20.00						
11	5-Jan	Lodging	89.00						
12	6-Jan	Food	9.00						
13	6-Jan	Food	3.50						
14	6-Jan	Food	11.02						
15	6-Jan	Food	78.44						

Figure 20-17: You can use array formulas to summarize data such as this in a dynamic crosstab table.

Array formulas were used to summarize this information into a handy table that shows the total expenses, by category, for each day. Cell F3 contains the following array formula, which was copied to the remaining 11 cells in the table:

```
{=SUM(IF($E3&F$2=DATES&CATEGORIES,AMOUNTS))}
```

These array formulas display the totals for each day, by category.

This formula operates similarly to the more simple one demonstrated in the preceding section. This formula has a few new twists, however. Rather than count the number of entries, the formula adds the appropriate value in the Amounts range. It does so, however, only if the row and column names in the summary table match the corresponding entries in the DATES and CATEGORIES ranges. It does the comparison by concatenating (using the & operator) the row and column names and comparing the resulting string to the concatenation of the corresponding DATES and CATEGORIES values. If the two match, the =SUM function kicks in and adds the corresponding value in the AMOUNTS range.

This technique can be customized, of course, to hold any number of different categories and any number of dates. You can eliminate the dates, in fact, and substitute people's names, departments, regions, and so on.

You also can cross-tabulate data by creating a pivot table. But, unlike a pivot table, using the procedure described here is completely dynamic (a pivot table must be updated if the data changes).

Pivot tables are discussed in Chapter 25.

Returning the Last Value in a Column

Suppose that you have a worksheet that you update frequently and need to determine the most recently entered value in a column. The following array formula returns the contents of the last nonempty cell in the first 500 rows of column A:

```
=INDIRECT(ADDRESS(MAX((ROW(1:500)*(A1:A500<>""))),COLUMN(A:A)))
```

You can modify this formula to work with a different column, and with a different number of rows in the column. To use a different column, change the column references from A to whichever column you need. To check more than 500 rows, change the two references to row 500.

Returning the Last Value in a Row

The following array formula is similar to the previous formula, but it returns the last nonempty cell in a row (in this case, row 1):

```
=INDIRECT(ADDRESS(1,(MAX((TRANSPOSE(ROW(1:256))*(1:1<>""))))))
```

To use this formula for a different row, change the first argument for the ADDRESS function to the new ADDRESS function and the 1:1 reference to correspond to the row.

A Single-Formula Calendar

The final array formula example is perhaps the most impressive. Figure 20-18 shows a monthly calendar that is calculated using a single array formula entered in B6:H11. This workbook includes a few additional bells and whistles. For example, you can choose the month and year to display by using dialog box controls that are inserted directly on the worksheet. When you change the month or year, the calendar is updated immediately.

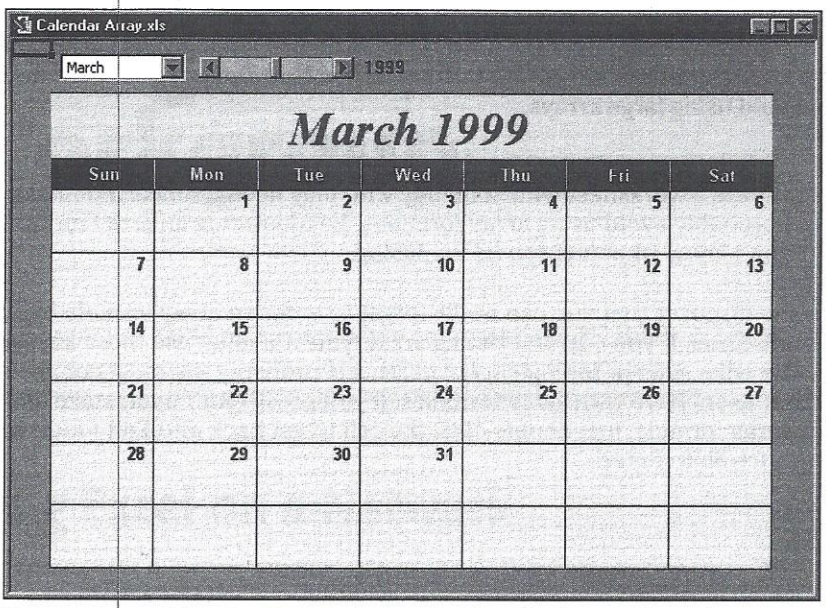


Figure 20-18: This calendar is calculated with a single array formula.

The array formula is as follows:

```
{=IF(MONTH(StartDate)<>MONTH(StartDate-StartDOW+Week*7+Weekday-
1),"",StartDate-StartDOW+Week*7+Weekday-1)}
```

This formula uses a few cell references (StartDate and StartDOW) and two named array constants, which are defined as follows:

```
Week:    ={0;1;2;3;4;5}
Weekday: ={1,2,3,4,5,6,7}
```

I leave it up to you to figure out how this works. Suffice it to say that it took more than a few minutes to develop.

The companion CD-ROM contains a workbook that uses the single-formula calendar.

Tips for Array Formulas

If you've followed along in this chapter, you probably understand the advantages of using array formulas. As you gain more experience with arrays, you undoubtedly will discover some disadvantages.

The primary problem with array formulas is that they slow your worksheet's recalculations, especially if you use large arrays. On a faster system, this may not be a problem. But if you have a slower system and speed is of the essence, you should probably avoid using large arrays.

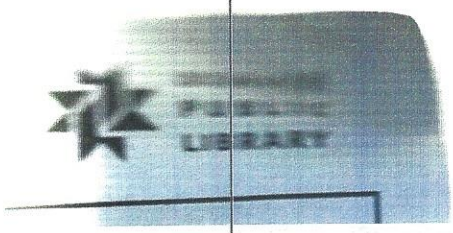
Array formulas are one of the least understood features of Excel. Consequently, if you plan to share a worksheet with someone who may need to make modifications, you should probably avoid using array formulas. Encountering an array formula when you don't know what it is can be confusing.

You may also discover that you can easily forget to enter an array formula by pressing Ctrl+Shift+Enter. If you edit an existing array, you still must use these keys to complete the edits. Except for logical errors, this is probably the most common problem that users have with array formulas. If you press Enter by mistake after editing an array formula, just double-click the cell to get back into Edit mode and then press Ctrl+Shift+Enter.

Summary

This chapter introduces the concept of *array formulas*, a special type of formula that operates on a group of cells. You can write an array formula by entering a single formula that performs an operation on multiple inputs and produces multiple results—with each result displayed in a separate cell. This chapter also presents several practical examples of array formulas.

✦ ✦ ✦



CHAPTER 21

Using Excel in a Workgroup

In This Chapter

If you use Excel on a standalone computer—a PC that's not connected to a network—you can skip this chapter, because it applies only to users who run Excel on a network.

Using Excel on a Network

A computer network consists of a group of PCs that are linked. A common type of network uses a *client-server model*, in which one or more PCs on the network act as dedicated *servers*, because they store files centrally and supply information, while user PCs are called *clients* (they use data in the centrally stored files on the server). Other networks are *peer-to-peer networks* that don't have a central server. Users on a network can perform the following tasks:

- Access files on other systems
- Share files with other users
- Share resources such as printers and fax modems
- Communicate with each other electronically

In many offices, networks now perform functions that formerly required a mainframe system and *dumb* terminals. Networks are usually less expensive, easier to expand, more manageable, and more flexible in terms of software availability than a mainframe system.

This chapter discusses the Excel features that are designed for network users.

File Reservations

Networks provide users with the ability to share information stored on other computer systems. Most networks have one

or more file servers attached. A file server stores files that members of a workgroup share. A network's file server may contain, for example, files that store customer lists, price lists, and form letters. Keeping these files on a file server has two major advantages:

- ✦ It eliminates the need to have multiple copies of the files stored locally on user PCs.
- ✦ It ensures that the file is always up to date; for example, if everyone makes changes to the same shared copy of a customer list, there's little likelihood that the portions of the list will be correct while other portions will be obsolete.

Some software applications are *multiuser applications*. Most database software applications, for example, enable multiple users to work simultaneously on the same database files. One user may be updating customer records in the database, while another is extracting records. But what if a user is updating a customer record and another user wants to make a change to that same record? Multiuser database software contains record-locking safeguards that ensure only one user at a time can modify a particular record.

Excel is *not* a multiuser application. When you open an Excel file, the entire file is loaded into memory. If the file is accessible to other users, you wouldn't want someone else to open a file that you've opened. If Excel allowed you to open and change a file that someone else on a network has already opened, the following scenario could happen.

Assume that your company keeps its sales information in an Excel file that is stored on a network server. Elaine wants to add this week's data to the file, so she loads it from the server and begins adding new information. A few minutes later, Albert loads the file to correct some errors that he noticed last week. Elaine finishes her work and saves the file. A while later, Albert finishes his corrections and saves the file. Albert's file overwrites the copy that Elaine saved, and her additions are gone.

This scenario *can't happen,* because Excel uses a concept known as *file reservation.* When Elaine opens the sales file, she has the reservation for the file. When Albert tries to open the file, Excel informs him that Elaine is using the file. If he insists on opening it, Excel opens the file as *read-only.* In other words, Albert can open the file, but he can't save it under the same name. Figure 21-1 shows the message that Albert receives if he tries to open a file that is in use by someone else.

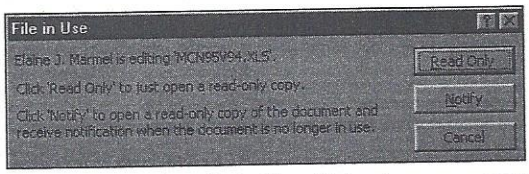


Figure 21-1: The File in Use dialog box appears if you try to open a file that someone else is using.

Albert has these three choices:

- ✦ **Select Cancel, wait a while, and try again.** He may call Elaine and ask her when she expects to be finished.
- ✦ **Select Read Only.** This lets him open the file to read it, but doesn't let him save changes to the same filename.
- ✦ **Select Notify, which opens the file as read-only.** Excel pops up a message when Elaine is finished using the file.

Figure 21-2 shows the message that Albert receives when the file is available.

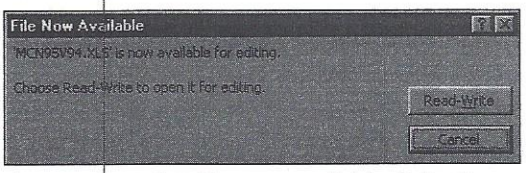


Figure 21-2: The File Now Available dialog box pops up with a new message when the file is available for editing.

Shared Workbooks

Although Excel isn't a multiuser application, it does support a feature known as *shared workbooks,* which enables multiple users to work on the same workbook simultaneously. Excel keeps track of the changes and provides appropriate prompts to handle conflict.

Appropriate Workbooks for Sharing

Although you can designate any workbook as a shared list, only certain workbooks contain information that is appropriate for sharing. The following are examples of workbooks that work well as shared lists:

- ✦ **Project tracking:** You may have a workbook that contains status information for projects. If multiple people are involved in the project, they can make changes and updates to the parts that are relevant.
- ✦ **Customer lists:** With customer lists, changes usually occur infrequently, but records are added and deleted.
- ✦ **Consolidations:** You may create a budget workbook in which each department manager is responsible for his or her department's budget. Usually, each department's budget appears on a separate sheet, with one sheet serving as the consolidation sheet.

Limitations of Shared Workbooks

If you plan to designate a workbook as shared, be aware that you cannot perform any of the following actions while sharing the workbook:

- ✦ Delete worksheets or chart sheets.
- ✦ Insert or delete a block of cells. However, you can insert or delete entire rows and columns.
- ✦ Merge cells.
- ✦ Define or apply conditional formats.
- ✦ Set up or change data-validation restrictions and messages.
- ✦ Insert or change charts, pictures, drawings, objects, or hyperlinks.
- ✦ Assign or modify a password to protect individual worksheets or the entire workbook.
- ✦ Create or modify pivot tables, scenarios, outlines, or data tables.
- ✦ Insert automatic subtotals.
- ✦ Make changes to dialog boxes or menus.
- ✦ Write, change, view, record, or assign macros. However, you can record a macro in a shared workbook that you store in another, unshared workbook.

Designating a Workbook As a Shared Workbook

To designate a workbook as a shared workbook, select Tools ⇨ Share Workbook. Excel displays the dialog box that is shown in Figure 21-3. This dialog box has two tabs: Editing and Advanced. In the Editing tab, select the check box to allow changes by multiple users and then click OK. Excel then prompts you to save the workbook.

When you open a shared workbook, the window's title bar displays [Shared]. If you no longer want other users to be able to use the workbook, remove the check mark from the Share Workbook dialog box and save the workbook.

Whenever you're working with a shared workbook, you can find out whether any other users are working on the workbook. Choose Tools ⇨ Share Workbook, and the Share Workbook dialog box lists the names of the other users who have the file open, as well as the time that each user opened the workbook.

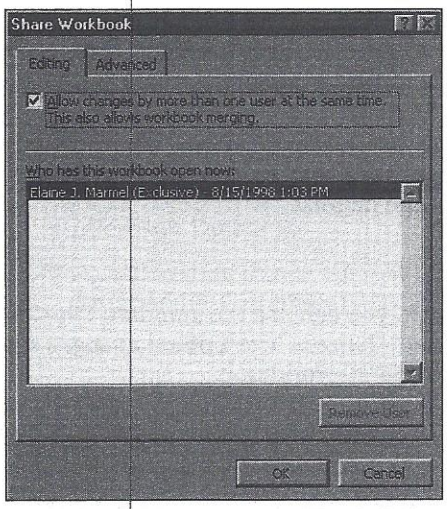


Figure 21-3: The Share Workbook dialog box lets you specify a workbook as a shared workbook.

Advanced Settings

Excel allows you to set options for shared workbooks. Select Tools ⇨ Share Workbook and click the Advanced tab to access these options (see Figure 21-4).

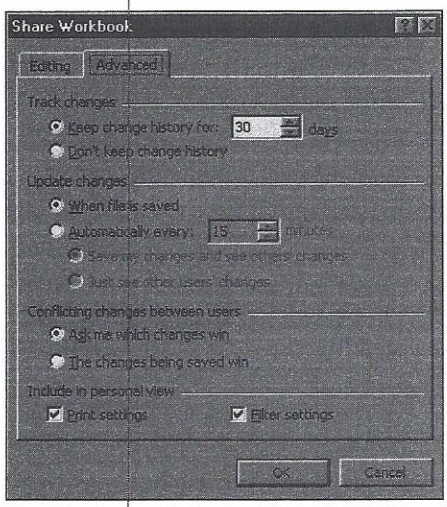


Figure 21-4: The Advanced tab of the Share Workbook dialog box.

Tracking changes

Excel can keep track of the workbook's changes — something known as *change history*. When you designate a workbook as a shared workbook, Excel automatically turns on the change history option, enabling you to view information about previous (and perhaps conflicting) changes to the workbook. You can turn off change history by selecting the option labeled Don't keep change history. You can also specify the number of days for which Excel tracks change history.

Updating changes

While you're working on a shared workbook, you can use the standard File➪Save command to update the workbook with your changes. The Update changes settings determine what happens when you save a shared workbook:

- **When file is saved:** You receive updates from other users when you save your copy of the shared workbook.
- **Automatically every:** Lets you specify a time period for receiving updates from other users of the workbook. You can also specify whether Excel should save your changes automatically, too, or just show you the changes made by other users.

Conflicting changes between users

As you may expect, multiple users working on the same file can result in some conflicts. For example, assume that you're working on a shared customer database workbook, and another user also has the workbook open. If you and the other user both make a change to the same cell, a conflict occurs. You can specify the manner in which Excel resolves the conflicts by selecting one of two options in the Advanced tab of the Share Workbook dialog box:

- **Ask me which changes win:** If you select this option, Excel displays a dialog box to let you determine how to settle the conflict.
- **The changes being saved win:** If you select this option, your changes always take precedence.

Include in personal view

The final section of the Advanced tab of the Share Workbook dialog box enables you to specify settings that are specific to your view of the shared workbook. You can choose to use your own print settings and your own data-filtering settings. If you don't place checks in these check boxes, you can't save your own print and filter settings.

Mailing and Routing Workbooks

Excel provides a few additional workgroup features. To use these features, your system must have one of the following items installed:

- ✦ Office 2000
- ✦ Microsoft Exchange
- ✦ A mail system that is compatible with MAPI (Messaging Application Programming Interface)
- ✦ Lotus cc:Mail
- ✦ A mail system that is compatible with VIM (Vendor Independent Messaging)

The procedures vary, depending on the mail system that you have installed; for this reason, discussions in the following sections are general in nature. For specific questions, consult your network administrator.

Mailing a Workbook As an E-mail Attachment

Electronic mail, or *e-mail*, is commonplace in most offices, and is an extremely. efficient means of communication. Unlike a telephone, e-mail doesn't rely on the recipient of the message being available when you want to send the message.

In addition to sending messages by e-mail, you can send complete files — including Excel workbooks. Like a growing number of software applications, Excel is *mail-enabled*, which means that you don't have to leave Excel to send a worksheet to someone by e-mail.

To send a copy of your workbook to someone on your network, select File ⇨ Send To ⇨ Mail Recipient (as Attachment). Excel creates an e-mail message with a copy of the workbook attached, using your default e-mail program; in Figure 21-5, Excel opened Outlook Express to send the workbook. You send this e-mail message the same way that you send any message — from your e-mail program. You also can send the message to multiple recipients, the same way that you send any e-mail message to multiple recipients.

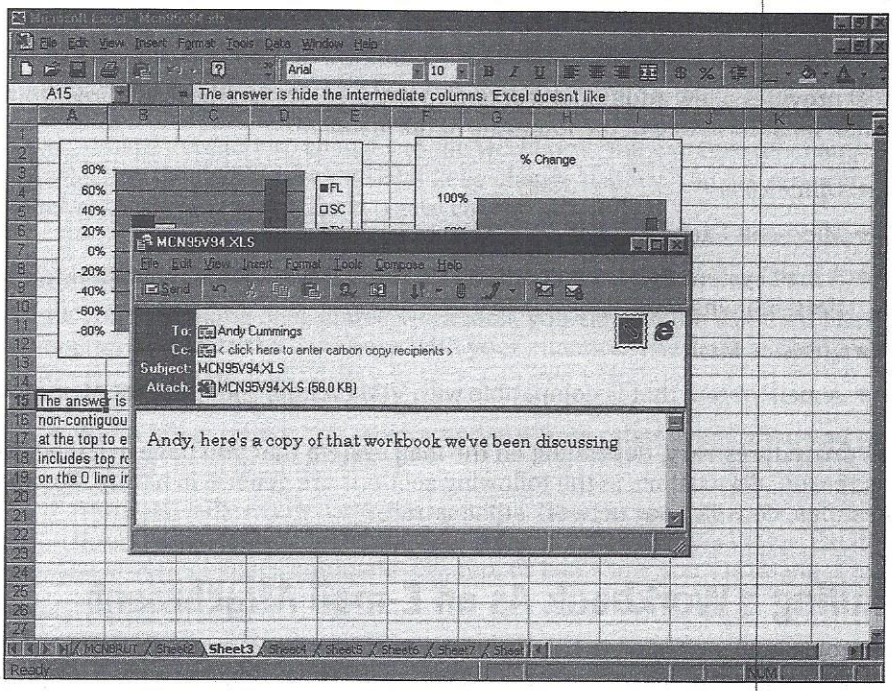


Figure 21-5: Sending a workbook as an attachment to an e-mail message.

When you send any file by using an e-mail program, you send a *copy* of the file. If the recipient makes changes to the notebook, the changes do not appear in your copy of the workbook.

Routing a Workbook to Others

If you choose File ⇨ Send To ⇨ Routing Recipient, Excel enables you to attach a routing slip to a workbook, similar to the one you see in Figure 21-6. Routing a workbook is most useful when you want the first person in the group to review (and possibly edit) the workbook and then send it to the next person on the list. For example, if you're responsible for your department's budget, you may need input from Alice — and her input may depend on Andy's input. You can set up the workbook and then route it to the others so that they can make their respective additions. When you set up the routing slip, you can tell Excel to return the workbook to you when the routing is finished.

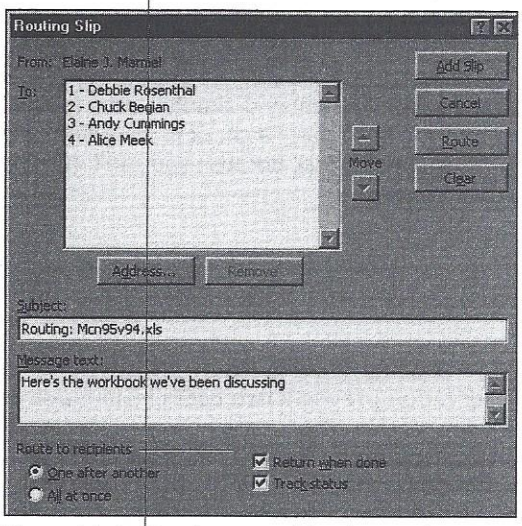


Figure 21-6: Routing a workbook.

When you route a workbook, you have the following two options:

- **Sequential routing:** Enables you to route the workbook sequentially to workgroup members. When the first recipient is finished, the workbook goes to the second recipient. When the second recipient is finished, the workbook goes to the third, and so on. When all recipients have received the workbook, it can be returned to you. Choose One after another at the bottom of the Routing Slip dialog box for this type of routing.
- **Simultaneous routing:** Enables you to route the workbook to all recipients at once. You receive a copy of the workbook from each recipient (not just one copy). This type of routing is useful if you want to solicit comments from a group of coworkers, and you want the responses back quickly (you don't want to wait until a single worksheet makes the circuit). Choose All at once at the bottom of the Routing Slip dialog box for this type of routing.

Click Route to route the workbook immediately. If you don't want to route immediately, click Add Slip. Later, when you're ready to route, choose File ➪ Send To ➪ Next Routing Recipient. Either choice places the workbook in the outgoing mail folder of your e-mail program. To actually route the workbook, open your e-mail program to send the message.

Whether you route or attach a workbook to an e-mail message, Excel uses your e-mail program. Since you can send a workbook to a number of people, either as an e-mail attachment or by using a routing slip, the distinction between the two methods lies in the distinction between sequential and simultaneous routing. If you choose simultaneous routing and you *don't* place a check in the Return when done check box, routing and attaching are identical, because you can't guarantee a reply to e-mail.

Summary

This chapter presents a basic overview of using Excel in a network environment. It explains how the concept of a file reservation prevents two users from modifying a workbook simultaneously. Excel's shared workbook feature, however, lets multiple users work on a single workbook at the same time. The chapter concludes with a discussion of mailing and routing workbooks.

✦ ✦ ✦

CHAPTER 22

Importing Data from Other Sources

In This Chapter

When you get right down to it, Excel can be described as a tool that manipulates data—the numbers and text that you use in a worksheet. But before you can manipulate data, it must be present in a worksheet. This chapter describes a variety of data-importing techniques.

An Overview of Importing

The following are the six basic ways to import data into Excel:

- Enter the data manually by typing values and text into cells
- Generate data by using formulas or macros
- Use Query (or a pivot table) to import data from an external database
- Import data from an HTML document on the Internet or a corporate intranet
- Copy data from another application by using the Windows Clipboard
- Import data from another (non-Excel) file

This chapter deals primarily with the last two methods: Clipboard copying and foreign-file importing.

Cross-Reference

Chapter 29 is somewhat related to this topic. It deals with linking to and from other applications and embedding objects. Querying external databases is covered in Chapter 24, and pivot tables are covered in Chapter 25. Chapter 30 discusses how Excel works with the Internet.

A Few Words About Data

Data is a broad concept that means different things to different people. Data is basically raw information that can come in any number of forms. For example, data can be numbers, text, or a combination. Most of what you do in Excel involves manipulating data in one way or another.

As computers become more commonplace, data is increasingly available in machine-readable formats (otherwise known as *files*). Not too long ago, major data suppliers provided printed reports to their clients. Now, data suppliers commonly offer a choice of formats: paper or disk.

Data that is stored in files can be in a wide variety of formats. Common file formats for distributing data include Lotus 1-2-3 files (WKS and WK1), dBASE files (DBF), and text files (which come in several varieties). Excel's file format is rather complex, and the format tends to change with every new version of Excel. Consequently, the Excel file format is not widely used for the general distribution of data.

The file format for Excel 2000 files is the same as the file format for Excel 97. However, if you use an Excel 2000 file in Excel 97, you will have access only to Excel 97 features.

As an Excel user, you need to understand the types of data that you can access either directly or indirectly.

File Formats Supported by Excel

Rarely does a computer user work with only one application or interact only with people using the same applications he or she uses. Suppose that you're developing a spreadsheet model that uses data from last year's budget, which is stored in your company's mainframe. You can request a printout of the data, of course, and manually enter it into Excel. If the amount of data isn't too large, this route may be the most efficient. But what if you have hundreds of entries to make? Your mainframe probably can't generate an Excel workbook file, but an excellent chance exists that it can send the report to a text file, which you can then import into an Excel worksheet. Potentially, you can save several hours of work and virtually eliminate data-entry errors.

As you know, Excel's native file format is an XLS file. In addition, Microsoft included the capability to read other file formats directly. For example, you can open a file that was created in several other spreadsheet products, such as Lotus 1-2-3 and Quattro Pro. Table 22-1 lists all the file formats that Excel can read (excluding its own file types).

To open any of these files, choose File ⇨ Open and select the file type from the drop-down list labeled Files of type (see Figure 22-1)to display only the files of the selected

type in the file list. If the file is a text file, Excel's Text Import Wizard appears, to help you interpret the file. The Text Import Wizard is discussed later in this chapter.

Table 22-1
File Formats Supported by Excel

File Type	*Description*
Text	Space delimited, tab delimited, and comma delimited
Lotus 1-2-3	Spreadsheet files generated by Lotus 1-2-3 for DOS Release 1.*x*, Release 2.*x*, Release 3.*x*, and 1-2-3 for Windows
Quattro Pro/DOS	Files generated by Novell's Quattro Pro for DOS spreadsheet
Microsoft Works 2.0	Files generated by Microsoft Works 2.0
dBASE	Database files in the DBF format
SYLK	Files generated by Microsoft's MultiPlan spreadsheet
Data Interchange Format	Files generated by the VisiCalc spreadsheet
HTML	Files developed for the World Wide Web
Quattro Pro for Windows	Files generated by Corel's Quattro Pro for Windows spreadsheet

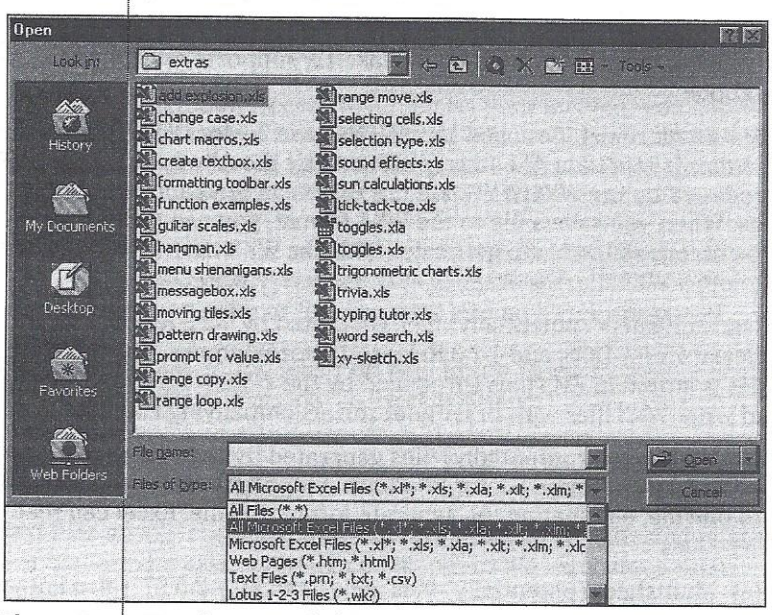


Figure 22-1: Use the Open dialog box to import a foreign file.

You should understand, however, that being able to read a file and translating it perfectly are two different matters. In some cases, you may encounter one or more of the following problems while reading a foreign file into Excel:

- ✦ Some formulas aren't translated correctly
- ✦ Unsupported functions aren't translated
- ✦ Formatting is incorrect
- ✦ Column widths are incorrect

When you open a file that wasn't produced by Excel, examine it carefully to ensure that Excel retrieved the data correctly.

The following sections discuss the various types of files that Excel can read. Each section discusses a file type and lists that file type's associated file extensions.

If a colleague sends you a file that Excel can't open, don't give up. Simply ask your colleague to save the spreadsheet in a format that Excel *can* read. For example, many applications can save files in 1-2-3 format, and most applications can export to a text file format.

Lotus 1-2-3 Spreadsheet Files

Lotus spreadsheets come in several flavors:

- ✦ **WKS files:** Single-sheet files used by 1-2-3 Release 1.*x* for DOS. Excel can read and write these files. If you export a workbook to a WKS file, Excel saves only the active worksheet, because 1-2-3 Release 1.*x* supports only one worksheet per workbook.
- ✦ **WK1 files:** Single-sheet files used by 1-2-3 Release 2.*x* for DOS. The formatting for these files is stored in ALL files (produced by the Allways add-in) or FM1 files (produced by the WYSIWYG add-in). Excel can read and write all of these file types. When you save a file to the WK1 format, you can choose which (if any) type of formatting file to generate. And, like WKS files, if you export a workbook to a WK1 file, Excel saves only the active worksheet.
- ✦ **WK3 files:** Multisheet (potentially) files generated by 1-2-3 Release 3.*x* for DOS, 1-2-3 Release 4.*x* for DOS, and 1-2-3 Release 1.*x* for Windows. The formatting for these files is stored in FM3 files (produced by the WYSIWYG add-in). Excel can read and write WK3 files with or without the accompanying FM3 file.
- ✦ **WK4 files:** Multisheet (potentially) files generated by 1-2-3 Release 4.*x* for Windows and 1-2-3 Release 5.*x* for Windows. Lotus combined formatting and data into one file, eliminating the separate formatting file. Excel can read and write these files.
- ✦ **1-2-3 files:** Multisheet (potentially) files generated by 1-2-3 97 (also known as Release 6) and 1-2-3 Millenium Edition (also known as Release 7). Excel can neither read nor write these files.

If you plan to import or export 1-2-3 files, I urge you to read the online Help for general guidelines and for specific types of information that may not be translated.

Excel evaluates some formulas differently from 1-2-3. To ensure complete compatibility when you work with an imported 1-2-3 file, choose Tools ⇨ Options, select the Transition tab, and then check the box labeled Transition Formula Evaluation.

Quattro Pro Spreadsheet Files

Quattro Pro files exist in several versions:

- **WQ1 files:** Single-sheet files generated by Quattro Pro for DOS Versions 1, 2, 3, and 4. Excel can read and write these files. If you export a workbook to a WQ1 file, Excel saves only the active worksheet.
- **WQ2 files:** Multisheet (potentially) files generated by Quattro Pro for DOS Version 5. Excel can neither read nor write this file format.
- **WB1 files:** Multisheet (potentially) files generated by Quattro Pro for Windows Versions 1 and 5 (there are no Versions 2 through 4). Excel can read (but not write) this file format.
- **WB2 files:** Multisheet (potentially) files generated by Quattro Pro for Windows Version 6. Excel can neither read nor write this file format.

Database File Formats

DBF files are single-table database files generated by dBASE and several other database programs. Excel can read and write DBF files up to and including dBASE 4.

If you have Microsoft Access installed on your system, you can take advantage of a feature that converts a worksheet list into an Access database file. To use this feature, you must install the Access Links add-in in Excel (you need your Office 2000 CD-ROM). Use the Data ⇨ Convert to MS Access command.

Excel can't read or write any other database file formats directly. If you install the Query add-in, however, you can use Query to access many other database file formats and then copy or link the data into an Excel worksheet.

See Chapter 24 for details on how to use Query to copy or link data from other database file formats into an Excel worksheet.

Text File Formats

Text files simply contain data—no formatting. The following relatively standard text file formats exist, although no standard file extensions exist:

- **Tab-delimited files:** Each line consists of fields that are separated by tabs. Excel can read these files, converting each line to a row and each field to a column. Excel also can write these files, using TXT as the default extension.
- **Comma-separated files:** Each line consists of fields that are separated by commas. Sometimes, text appears in quotation marks. Excel can read these files, converting each line to a row and each field to a column. Excel can also write these files, using CSV as the default extension.
- **Space-delimited files:** Each line consists of fields that are separated by spaces. Excel can read these files, converting each line to a row and each field to a column. Excel also can write these files, using PRN as the default extension.

If you want your exported text file to use a different extension, specify the complete filename and extension in quotation marks. For example, saving a workbook in comma-separated format normally uses the CSV extension. If you want your file to be named output.txt (with a TXT extension), enter **"output.txt"** in the File name box in the Save As dialog box.

When you attempt to load a text file into Excel, the Text Import Wizard kicks in to help you specify how you want Excel to retrieve the file (discussed in detail later in the chapter).

HTML Files

Excel can read and save files in HTML (Hypertext Markup Language) format, a file format that is used on the World Wide Web. And, through the use of XML (Extensible Markup Language), HTML files retain all document properties, including fonts and formatting.

Using Excel, you can edit any Excel document from within a Web browser. While you are viewing a page that was created in an Office application, such as Excel, click the Edit button on the browser's toolbar. Office opens the document in the application that was used to create it. You can then edit the Web page and resave it in any of the file formats that the application supports or in HTML.

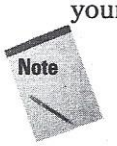
Other File Formats

The following are two other types of file formats that you will rarely encounter; I haven't seen a DIF file in ages, and I've never seen a SYLK file.

- **Data Interchange Format (DIF):** Used by VisiCalc. Excel can read and write these files.
- **Symbolic Link (SYLK):** Used by MultiPlan. Excel can read and write these files.

Using the Clipboard to Get Data

Using the Windows Clipboard is a another method of importing data into your worksheet. The process involves selecting data from another application and copying the data to the Clipboard. Then, you reactivate Excel and paste the information to the worksheet. The exact results that you get can vary quite a bit, depending on the type of data that you copied and the Clipboard formats that it supports. Obviously, you must have a copy of the other application installed on your system.



If you copy information from another Office application, you use the Office Clipboard, not the Windows Clipboard. The Office Clipboard supports copying and pasting of all formats used in all Office applications.

About the Clipboard

As you read in Chapter 8, Office 2000 provides Windows 95 or Windows 98 with two clipboards. The original Windows Clipboard remains; whenever you cut or copy information from a Windows program, Windows stores the information on the Windows Clipboard, which is an area of memory. Each time that you cut or copy information, Windows replaces the information previously stored on the Clipboard with the new information that you cut or copied. The Windows Clipboard can store data in a variety of formats. Because Windows manages it, information on the Windows Clipboard can be pasted to other Windows applications, regardless of where it originated. Normally, you can't see information stored on the Windows Clipboard (nor would you want to).

To view the Windows Clipboard contents, you can run the Clipboard Viewer program, which comes with Windows. The Clipboard Viewer may or may not be installed on your system (it is not installed by default). You can use the Clipboard Viewer to view only the last piece of information that you copied to the Office Clipboard.

When you copy or cut data to the Clipboard, the source application places one or more formats on the Clipboard along with the data. Different applications support different Clipboard formats. When you paste Clipboard data into another application, the destination application determines which format it can handle and typically selects the format that either provides the most information or is appropriate for where you are pasting it. If you view cells copied from Excel in the Clipboard Viewer, by default, you'll see a row/column reference rather than the actual information that you copied (see Figure 22-2). In some cases, you can use the Display command in the Clipboard Viewer application to view the Clipboard data in a different format. For example, you can display a range of cells from Excel as a picture, bitmap, text, OEM text, or a DIB bitmap. Figures 22-3, 22-4, 22-5, and 22-6, show examples of the same Excel range as it appears in the Clipboard Viewer when using different Display formats.

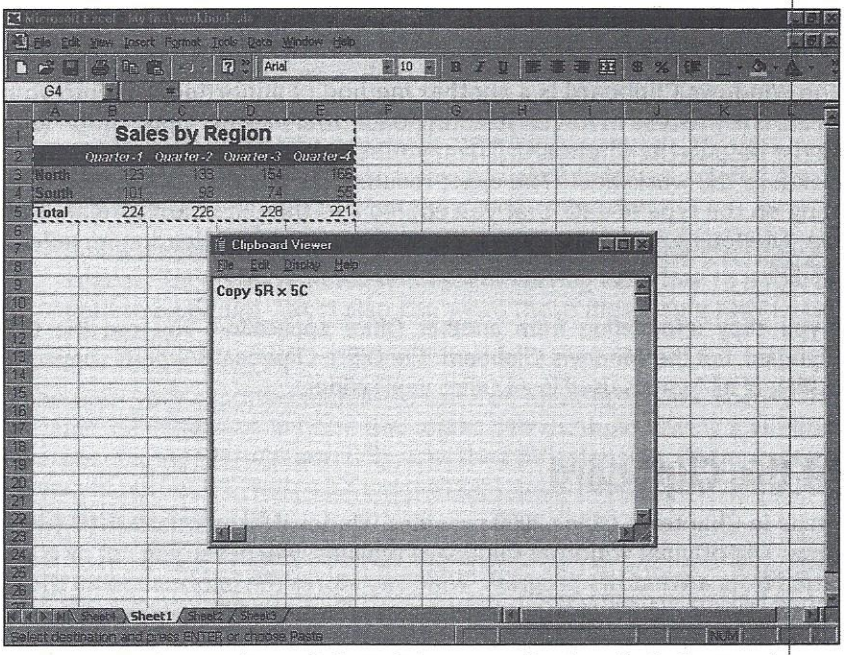


Figure 22-2: The Windows Clipboard Viewer application displaying Excel 2000 data in default format, Display Text.

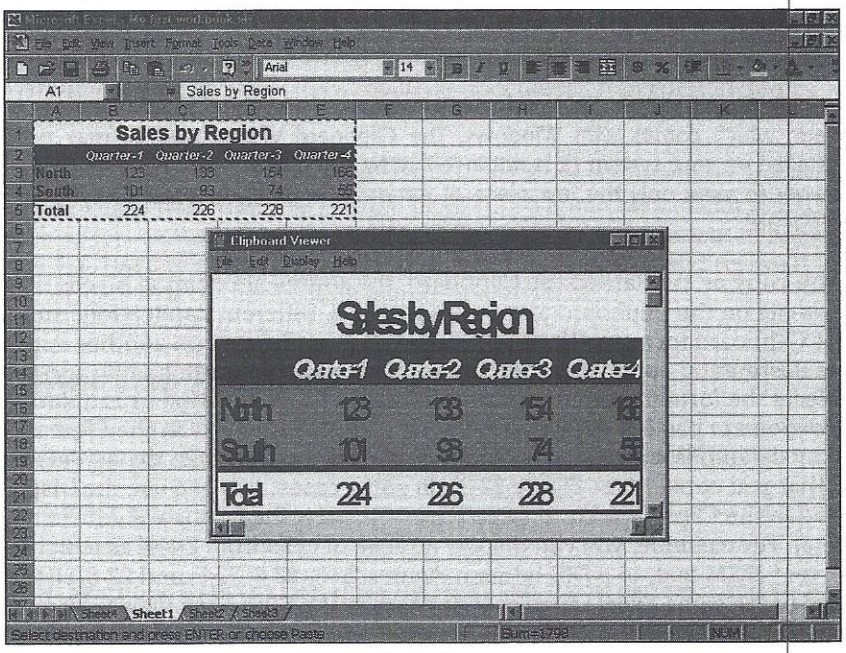


Figure 22-3: The same data in Picture format.

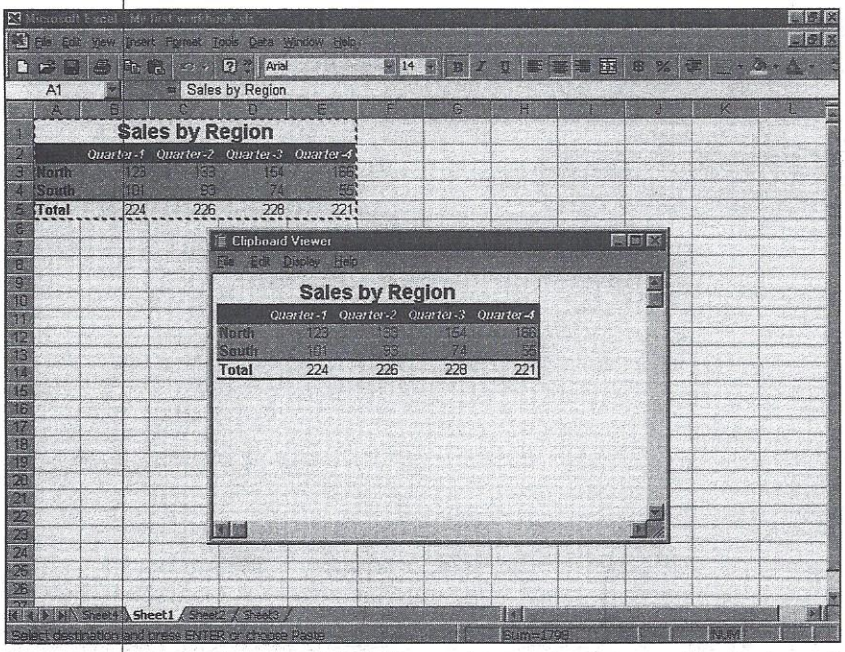


Figure 22-4: Both Bitmap and DIB Bitmap format closely resemble the formatted appearance of the data in Excel.

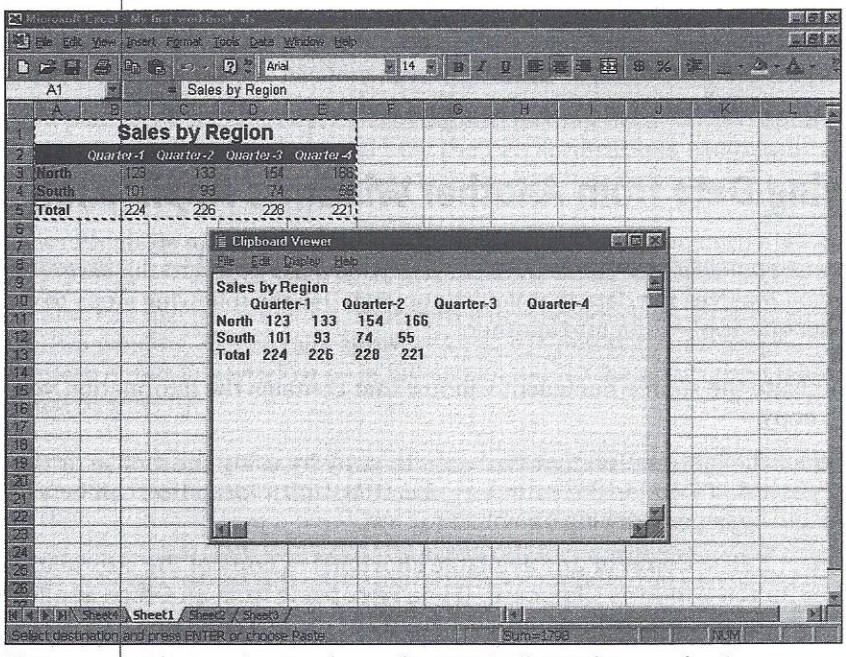


Figure 22-5: The Text format shows the text similar to the way that it appears in the Notepad.

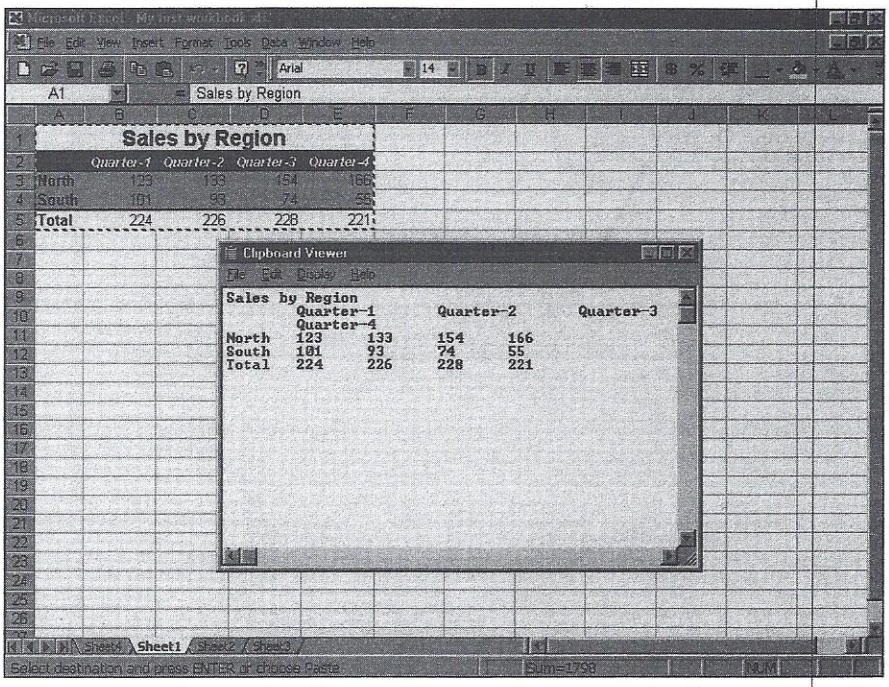


Figure 22-6: The OEM text format.

Importantly, the format that you select in the Clipboard Viewer *doesn't* affect how Excel copies the data. In some cases, however, you can use Excel's Edit ⇨ Paste Special command to select alternate methods of pasting the data.

Copying Data from Another Windows Application

Copying data from one Windows application to another is quite straightforward. The *source application* contains the data that you're copying, and the *destination application* receives the data that you're copying. Use the following steps to copy data from one application into another:

1. Activate the source document window that contains the information you want to copy.
2. Select the information that you want to copy by using the mouse or the keyboard. If Excel is the source application, this information can be a cell, range, chart, or drawing object.
3. Select Edit ⇨ Copy (or any available shortcut). A copy of the information is sent to the Windows Clipboard. If you're copying from an Office application, a copy of the information is also sent to the Office Clipboard.
4. Activate the destination application. If it isn't open, you can start it without affecting the contents of the Clipboard.

5. Move to the position to which you want to paste in the destination application.
6. Select Edit ⇨ Paste from the menu in the destination application. If the Clipboard contents aren't appropriate for pasting, the Paste command is grayed (not available).

In Step 3, you also can select Edit ⇨ Cut from the source application menu. This step erases the selection from the source application after it's placed on the Clipboard.

Many Windows applications use a common keyboard convention for the Clipboard commands. Generally, this technique is a bit faster than using the menus, because these keys are adjacent to each other. The shortcut keys and their equivalents are the following:

Ctrl+C	Edit ⇨ Copy
Ctrl+X	Edit ⇨ Cut
Ctrl+V	Edit ⇨ Paste

You need to understand that Windows applications vary in how they respond to data that you paste from the Clipboard. If the Edit ⇨ Paste command isn't available (it is grayed on the menu) in the destination application, the application can't accept the information from the Clipboard. If you copy a table from Word for Windows to Excel, the data translates into cells perfectly—complete with formatting. Copying data from other applications may not work as well; for example, you may lose the formatting, or you may end up with all the data in a single column rather than in separate columns. As discussed later in this chapter, you can use the Convert Text to Columns Wizard to convert this data into columns.

If you plan to do a great deal of copying and pasting between two applications, experiment until you understand how the two applications can handle each other's data.

Copying Data from a Non-Windows Application

You also can use the Windows Clipboard with non-Windows applications running in a DOS window. As you may know, you can run non-Windows programs from Windows either in a window or in full-screen mode (the application takes over the complete screen).

When you're running a non-Windows application in Windows, you can press Alt+Print Screen to copy the entire screen to the Clipboard. The screen contents can then be pasted into a Windows application (including Excel). To copy only part of the screen, you must run the application in a window: press Alt+Enter to toggle between full-screen mode and windowed mode. You can then click the Control menu, choose Edit ⇨ Mark, and select text from the window. This window may or may not have a toolbar displayed. If it does not, follow these steps:

1. Right-click the title bar and select the Toolbar option.

2. Click the Mark tool and select the text to copy.
3. Click the Copy tool to copy the selected text to the Clipboard.
4. Activate Excel.
5. Select Edit ➪ Paste to copy the Clipboard data into your worksheet.

Figure 22-7 shows Quattro Pro running in a DOS window. Some text is selected.

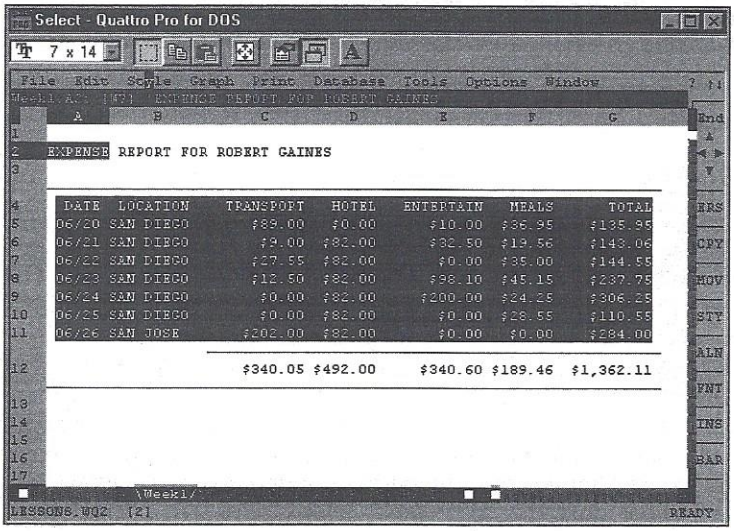


Figure 22-7: Copying data from Quattro Pro for DOS.

If you use this technique and copy to Excel, the information is pasted as text in a single column. In other words, even if you copy information from neatly formatted columns, it's all pasted into a single column in Excel. But don't fret—you can use Excel's Convert Text to Columns Wizard to convert this data into columns.

You're limited to copying one screen of information at a time—you can't scroll the DOS application while you're selecting text.

Importing Text Files

Text files (sometimes referred to as ASCII files) are usually considered to be the lowest-common-denominator file type. Such files contain only data, with no formatting. Consequently, most applications can read and write text files. So, if all else fails, you can probably use a text file to transfer data between two applications that don't support a common file format. Because text files are so commonly used, this entire section is devoted to discussing them and explaining how to use Excel's Text Import Wizard.

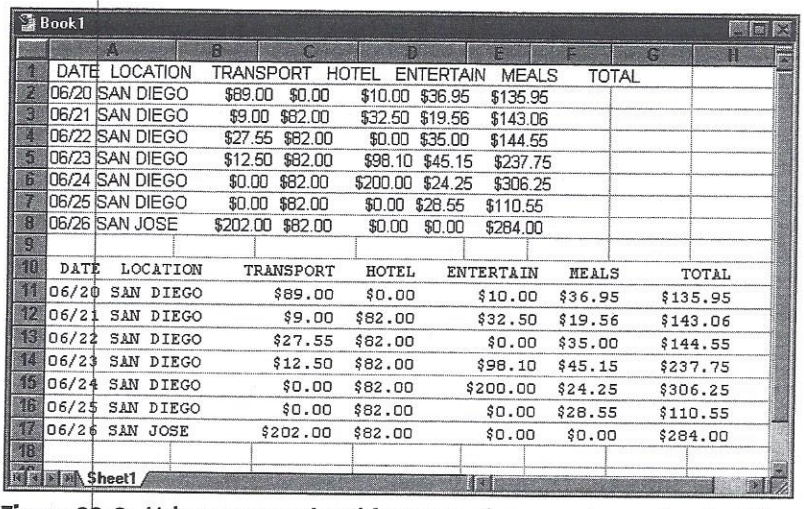
About Text Files

You may find it helpful to think of some text files in terms of a database table. Each line in the text file corresponds to a database record, and each record consists of a number of fields. In Excel, each line (or record) is imported to a separate row, and each field goes into a separate column. Text files come in two types: delimited and nondelimited.

Text files consist of plain text and end-of-line markers. *Delimited* text files use a special character to separate the fields on each line—typically a comma, a space, or a tab (but occasionally, you'll see other delimiters used). In addition, text is usually (but not always) enclosed in quotation marks.

Nondelimited files don't contain a special field-separator character. Often, however, each field is a fixed length, enabling you easily to break each line of text into separate columns. When you view a nondelimited file, the data often appears to be in columns.

If you use a proportional font, such as Arial or Times Roman, the fields of text file may appear to not line up, although they actually do. In proportional font sets, each character uses a different amount of horizontal space. For best results, use a nonproportional font, such as Courier New, when working with text files. Excel uses Courier New in its Text Import Wizard dialog box. Figure 22-8 shows the same text displayed in Arial and Courier New fonts.



Book1

	A	B	C	D	E	F	G	H
1	DATE	LOCATION	TRANSPORT	HOTEL	ENTERTAIN	MEALS	TOTAL	
2	06/20	SAN DIEGO	$89.00	$0.00	$10.00	$36.95	$135.95	
3	06/21	SAN DIEGO	$9.00	$82.00	$32.50	$19.56	$143.06	
4	06/22	SAN DIEGO	$27.55	$82.00	$0.00	$35.00	$144.55	
5	06/23	SAN DIEGO	$12.50	$82.00	$98.10	$45.15	$237.75	
6	06/24	SAN DIEGO	$0.00	$82.00	$200.00	$24.25	$306.25	
7	06/25	SAN DIEGO	$0.00	$82.00	$0.00	$28.55	$110.55	
8	06/26	SAN JOSE	$202.00	$82.00	$0.00	$0.00	$284.00	
9								
10	DATE	LOCATION	TRANSPORT	HOTEL	ENTERTAIN	MEALS	TOTAL	
11	06/20	SAN DIEGO	$89.00	$0.00	$10.00	$36.95	$135.95	
12	06/21	SAN DIEGO	$9.00	$82.00	$32.50	$19.56	$143.06	
13	06/22	SAN DIEGO	$27.55	$82.00	$0.00	$35.00	$144.55	
14	06/23	SAN DIEGO	$12.50	$82.00	$98.10	$45.15	$237.75	
15	06/24	SAN DIEGO	$0.00	$82.00	$200.00	$24.25	$306.25	
16	06/25	SAN DIEGO	$0.00	$82.00	$0.00	$28.55	$110.55	
17	06/26	SAN JOSE	$202.00	$82.00	$0.00	$0.00	$284.00	
18								

Sheet1

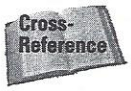
Figure 22-8: Using a proportional font may obscure columns in a text file.

Excel is quite versatile when importing text files. If each line of the text file is identically laid out, importing is usually problem-free. But if the line contains mixed information, you may need to do some additional work to make the data usable.

For example, you create text files in some programs by sending a printed report to a disk file rather than to the printer. These reports often have extra information, such as page headers and footers, titles, summary lines, and so on.

Using the Text Import Wizard

Prior versions of Excel treated importing text files differently from other types of database information. In Excel 2000, if you use the technique described in this section, you'll create a Text File Query, which you can refresh in the same way that you refresh Database and Web queries. This new feature will make easier the lives of those who need to regularly import text files, because they won't need to "set up" the import each time. When you want to update the Excel file that you create by importing a text file, choose Data ⇨ Refresh Data. Highlight the text file that you originally imported and click the Import button. Excel automatically updates the Excel version of the file with any new data that may appear in the text file.



For more information on Database queries, see Chapter 24. For more information on Web queries, see Chapter 30.

To import a text file into Excel, choose Data ⇨ Get External Data ⇨ Import Text File. In the Import Text File dialog box, navigate to the folder containing the file that you want to import. The dialog box then displays text files that have an extension of TXT. If the text file that you're importing has a different extension, select the All Files option. Or, you can enter the filename directly into the File name box, if you know the file's name.

Excel displays its Text Import Wizard, a series of interactive dialog boxes in which you specify the information that Excel requires to break the lines of the text file into columns. You can truly appreciate this time-saving feature if, in a previous life, you struggled with the old data-parsing commands that are found in other spreadsheet programs and older versions of Excel.

Text Import Wizard: Step 1 of 3

Figure 22-9 shows the first of three Text Import Wizard dialog boxes. In the Original data type section, verify the type of data file (Excel almost always guesses correctly). You also can indicate the row that Excel should use to start importing. For example, if the file has a title in the first row, you may want to skip the first line.

Notice that you can preview the file at the bottom of the dialog box, using the scrollbars to view more of the file. If the characters in the file don't look right, you may need to change the File Origin; this determines which character set to use (in many cases, it doesn't make any difference). After you finish with this step, click the Next button to move to Step 2.

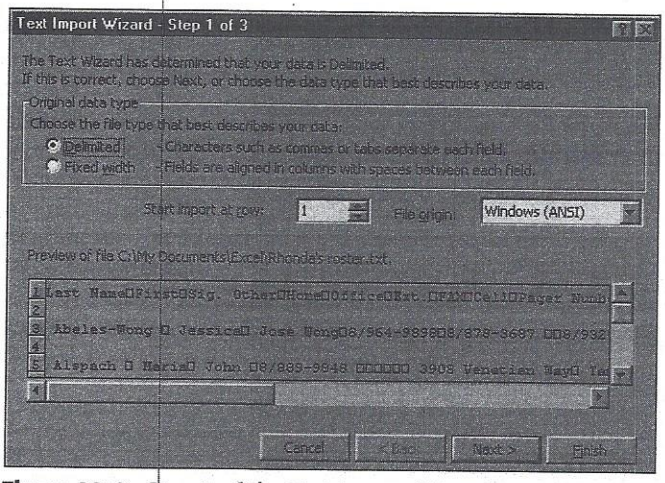


Figure 22-9: Step 1 of the Text Import Wizard.

Text Import Wizard: Step 2 of 3

The dialog box that you see for Step 2 of the Text Import Wizard varies, depending on your choice in the Original data type section in Step 1. If you selected Delimited, you see the dialog box shown in Figure 22-10. You can specify the type of delimiter, the text qualifier, and whether to treat consecutive delimiters as a single delimiter; choosing to treat consecutive delimiters as a single delimiter tells Excel to skip empty columns. The Data preview section displays vertical lines to indicate how Excel will break up the fields. The Data preview section changes as you make choices in the dialog box.

If you selected Fixed width, you see the dialog box shown in Figure 22-11. At this point, Excel attempts to identify the column breaks and displays vertical break lines to represent how it will break fields apart into columns. If Excel guesses wrong, you can move the lines, insert new ones, or delete lines that Excel proposes. You'll see instructions in the dialog box.

Text Import Wizard: Step 3 of 3

Figure 22-12 shows the last of the three Text Import Wizard dialog boxes. In this dialog box, you can select individual columns and specify the formatting to apply (General, Text, or Data). You also can specify columns to skip—they aren't imported. If you click the Advanced button, you'll see the dialog box shown in Figure 22-13, in which you can specify characters to use as decimal and thousands separators. When you're satisfied with the results, click Finish. Excel prompts you for the starting cell location for the imported data; when you click OK, Excel imports the data and displays the External Data toolbar, which helps you to work with the imported text file (see Figure 22-14). For example, if you click the Data Range Properties tool, you see the External Data Range Properties dialog box, shown in Figure 22-15, which you can use to change how Excel treats the imported file.

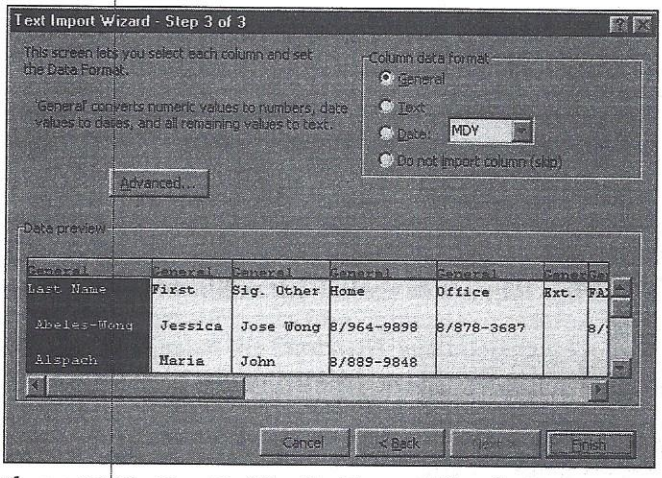


Figure 22-12: Step 3 of the Text Import Wizard.

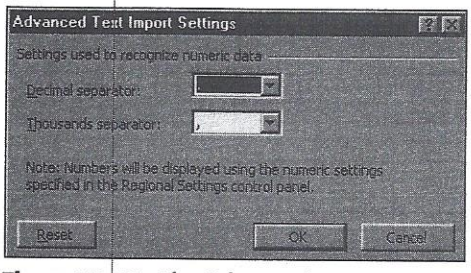
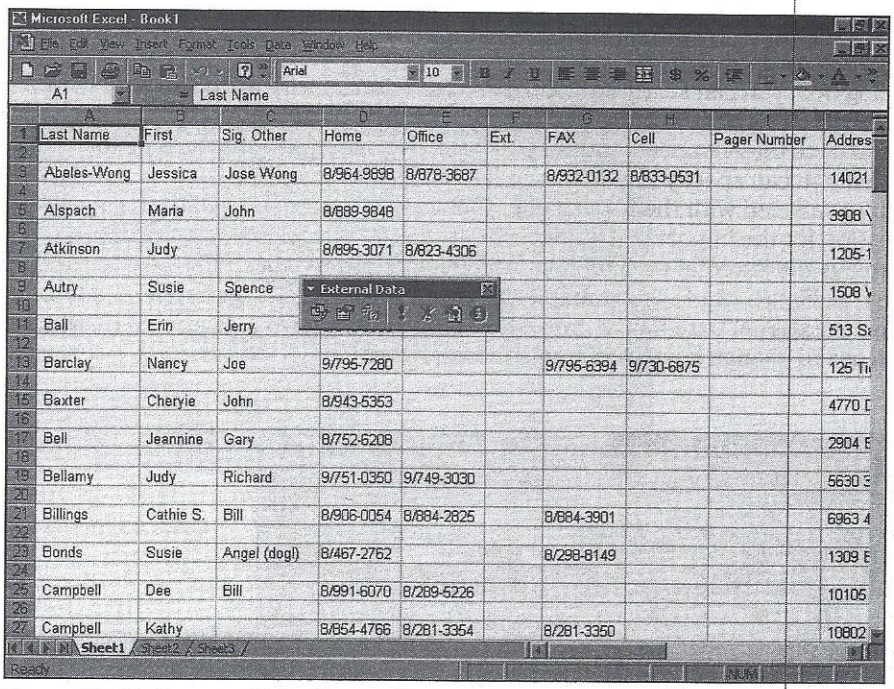


Figure 22-13: The Advanced Text Import Settings dialog box.



	A	B	C	D	E	F	G	H	I	J
1	Last Name	First	Sig. Other	Home	Office	Ext.	FAX	Cell	Pager Number	Addres
2										
3	Abeles-Wong	Jessica	Jose Wong	8/964-9898	8/878-3687		8/932-0132	8/833-0531		14021
4										
5	Alspach	Maria	John	8/889-9848						3908 \
6										
7	Atkinson	Judy		8/895-3071	8/823-4306					1205-1
8										
9	Autry	Susie	Spence							1508 \
10										
11	Ball	Erin	Jerry							513 S
12										
13	Barclay	Nancy	Joe	9/795-7280			9/795-6394	9/730-6875		125 Ti
14										
15	Baxter	Cheryie	John	8/943-5353						4770 [
16										
17	Bell	Jeannine	Gary	8/752-6208						2904 E
18										
19	Bellamy	Judy	Richard	9/751-0350	9/749-3030					5630 S
20										
21	Billings	Cathie S.	Bill	8/906-0054	8/684-2825		8/684-3901			6963 4
22										
23	Bonds	Susie	Angel (dogl)	8/467-2762			8/298-8149			1309 E
24										
25	Campbell	Dee	Bill	8/991-6070	8/269-5226					10105
26										
27	Campbell	Kathy		8/854-4766	8/281-3354		8/281-3350			10802

Figure 22-14: Imported text file.

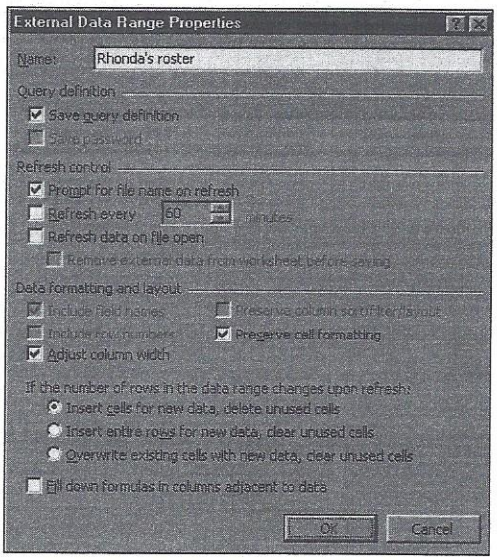


Figure 22-15: Use this dialog box to change the way that Excel treats the imported file.

If the results aren't what you expect, close the workbook and try again (text importing often involves trial and error). Don't forget that you can scroll the Data Preview window to make sure that all the data is converted properly. With some files, however, importing all the data properly is impossible. In such cases, you may want to import the file as a single column of text and then break lines into columns selectively. The procedure for doing this is discussed in the next section.

Using the Text to Columns Wizard

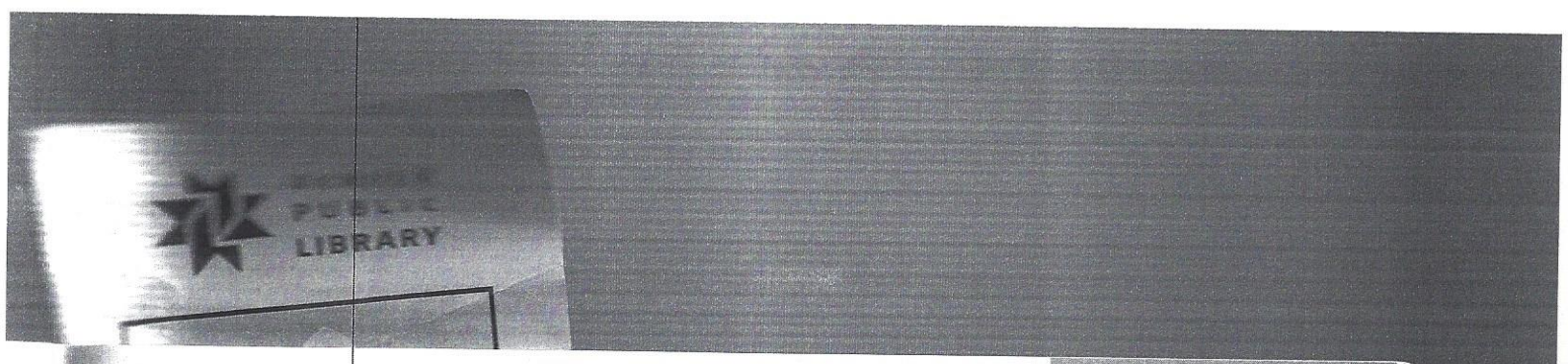
Excel can parse text that is stored in a column. Start by selecting the text (in a single column). Then, choose Data ⇨ Text to Columns, and Excel displays the first of three Text to Columns Wizard dialog boxes. These dialog boxes are identical to those used for the Text Import Wizard, except that the title bar text is different.

Unfortunately, you can't use the Data ⇨ Text to Columns command on a multiple selection; this would be quite handy for parsing imported files with several different layouts. Even worse, you can't use the Edit ⇨ Repeat command to repeat the Text to Columns command.

Summary

This chapter identifies the various sources for getting data into Excel: entering data manually, generating data from formulas or macros, using Query or pivot tables, copying data using the Clipboard, and importing foreign files (including text files) into Excel. The chapter focuses on Clipboard operations and file importing.

✦ ✦ ✦



CHAPTER 23

Working with Lists

In This Chapter

Research conducted by Microsoft indicates that Excel is frequently used to manage lists, or *worksheet databases.* This chapter covers list management and demonstrates useful techniques that involve lists.

What Is a List?

A list is essentially an organized collection of information. More specifically, a list consists of a row of headers (descriptive text), followed by additional rows of data, which can be values or text. You may recognize this as a database table—which is exactly what it is. Beginning with Excel 5, Microsoft uses the term *list* to refer to a database stored in a worksheet and the term *database* to refer to a table of information stored in an external file. To avoid confusion, I adhere to Microsoft's terminology.

Cross-Reference I cover external database files in Chapter 24.

Figure 23-1 shows an example of a list in a worksheet. This particular list has its headers in row 1 and has 10 rows of data. The list occupies four columns. Notice that the data consists of several different types: text, values, and dates. Column C contains a formula that calculates the monthly salary from the value in column B.