# Chapter 1: Getting with the Program

## In This Chapter

- **Exploring what you'll find in this book**
- **Managing menus and toolbars**
- **Creating your first Office program (it's easier than you think)**
- **Customizing the keyboard**
- **Getting programming help online**

Over the years, Office has evolved. As the result of hundreds of focus groups, ergonomic studies, user feedback, and hard-won experience, the Office design teams have come up with a highly effective suite of applications.

One of Office's strong points over the years has been its considerable depth. You can find literally thousands of features within the Office applications, yet the surface that you interact with can be as smooth and simple as you wish. You can even hide the toolbars and menu bars.

Put another way, Office applications are highly customizable. Throughout this book, you discover ways to manage and exploit Office 2003 to take it to a new level of efficiency. You see how to write programs that make your work easier as well as how to build utilities that facilitate communication between Office applications and automate other common business tasks. I also show you hundreds of other useful techniques and tools.

Most new computers ship with Office, which is also used in nearly every business today. What these businesses have in common is an ongoing effort to improve their efficiency. And in many cases, developing or automating Office applications is one of the most effective ways to increase workplace productivity. Many workers know what they wish they could do — and this book shows them how to do it.

*Office 2003 Application Development All-in-One Desk Reference For Dummies* covers all the new features in Office 2003 and demonstrates how developers can best exploit them. Many of these features are designed to improve workflow and facilitate better communication between workers — just the sort of

goals that Office developers want to achieve. You'll find everything you need to know to make Office 2003 an effective, valuable, and customized workplace engine.

For example, InfoPath simplifies interaction with all kinds of data sources: everything from unformatted lists to legacy databases. SharePoint assists developers in building an automated collaborative environment. And eXtensible Markup Language (XML) as well as related technologies, such as Web Services, underlies many of the improvements in Office 2003. Among many other topics, this book explains precisely how to take advantage of XML's promise with simple, no-nonsense examples. You'll understand exactly how to leverage your current work and communication patterns by using the new and powerful data sharing techniques available in Office 2003.

No significant Office topic is ignored here. Read on to discover how to use classic but important features such as Visual Basic for Applications (VBA). And I cover all the latest developments such as Smart Documents, Access 2003 Developer Extensions, and the new security features. Anyone interested in building intelligent business applications will find the solutions they're looking for here.

# Modifying the User Interface

This chapter starts things off with an introduction to some relatively easy modifications that you can make to Office 2003 applications. (No point jumping immediately into the deep end of heavy-duty programming; there's time enough for all that in subsequent chapters.) And although these modifications are on the simpler side, some of the techniques that I describe in this chapter are powerful, new, or both.

I show you various ways of manipulating the user interface, the surface that you work on when using Office applications. If you're already an Office guru, you might want to skim this chapter to search for techniques you don't yet know. Less-experienced readers are likely to find many ideas in this chapter that are of immediate practical use, such as hiding a new Help feature or modifying and editing menus.

You will find a bit of programming in this chapter, too, but it's not very advanced, and you're not even expected to understand it at this point. You can just benefit from the exercise, and you might find the results (such as quickly turning toolbars on and off) a valuable addition to your bag of tricks.

## Turning off mini help

Begin by seeing how to get rid of that new little Office 2003 Help field. It seems as if Microsoft introduced a new, cute Help feature. Remember the little paper clip fellow (Clippit) that started annoying many people a few

years back? Most people find the animated Office Assistant rather bothersome, not to mention unprofessional looking. At least it's easy enough to turn off that annoying paper clip by just deselecting Office Assistant on the Help menu.

Now to get rid of that new Help field, located by default at the upper-right corner of Office applications such as Word and Access, as you can see in Figure 1-1.


**Figure 1-1:** Type Help questions here or hide Help.

What I don't like about this feature is that anyone can see your last Help request, which could be embarrassing. Personally, I don't want people seeing the kind of help that I last requested. It lets them know what I didn't know. Your last question stays up there for all to see, even after you've closed the Help pane. And even though there's no obvious way to make this little feature go away, I know the easy secret.


Before I lead you through this example, note something important about the Customize dialog box. When it's open, Office 2003 applications freeze and wait to see whether you're modifying something. All the menus and toolbars are loosened, so you can drag and drop items from the dialog box onto toolbars or remove items by dragging them off toolbars and dropping them into the document workspace.

Here are the steps for removing the Help field:

1. **Choose Tools➪Customize.**
2. **Click the Toolbars tab of the Customize dialog box.**
3. **Right-click the little Help field.**

   You see a check box with a check mark in it.
4. **Remove the check and then close the Customize dialog box.**

   The discomforting little critter won't ever advertise your personal shortcomings again.

If you want this feature back, just repeat these steps, but mark the check box to select it.


Making this change in Word won't get rid of the Help field in other Office 2003 applications; you have to turn it off in each application.

## Modifying menus

Menus can be adjusted to suit your needs. You can move the menu bar itself the same way that you move toolbars: Just drag the dotted line on the left or top of the menu bar and drop the menu elsewhere on the screen. To switch between long and short menus (short menus display only the most frequently used options), choose Tools➪Customize➪Options, and then select the Always Show Full Menus check box.

To modify a menu's location on the menu bar, follow these steps:

1. **Choose Tools➪Customize.**
2. **While the Customize dialog box is open, you're free to drag around the menu headings on the menu bar, reorganizing them any way you wish.**

   If you want to remove a menu heading entirely, drag it away from the menu bar and drop it somewhere in the document.

## Editing menus

The contents of menus can be modified, too. To modify the order of items within a menu, follow these steps:

1. **Choose Tools➪Customize.**
2. **In the Customize dialog box, click the Commands tab.**
3. **Click the Rearrange Commands button.**

   The Rearrange Commands dialog box opens, as shown in Figure 1-2.

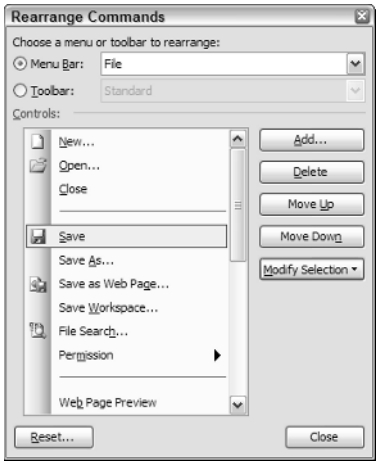


**Figure 1-2:** Add, delete, or adjust the order of commands here.

4. **Select a command you want to rearrange; then click the Move Up or Move Down buttons.**

Click the Modify Selection button in Figure 1-2 to rename menu items, change their icon, and otherwise manipulate them to suit yourself, as shown in Figure 1-3.

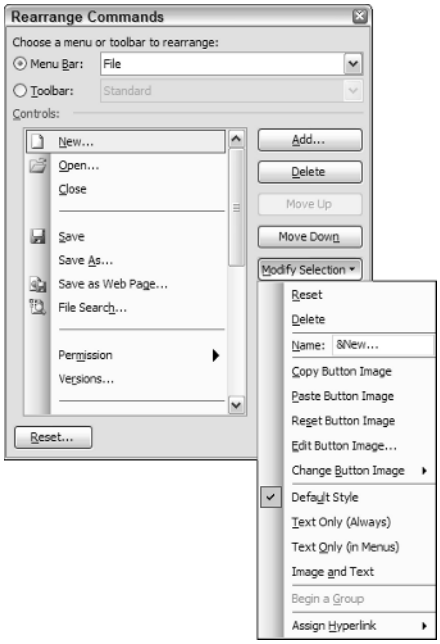


**Figure 1-3:** Here's where you can really take control of your menus.

## Creating your own menus

You can even create a new menu of your personal favorite features. Just follow these steps:

1. **Choose Tools⇨Customize and then click the Commands tab.**
2. **Click New Menu in the Categories list.**
3. **Drag the new menu icon from the Customize dialog box and drop it on the menu bar.**
4. **Right-click the new menu to name it whatever you want.**

   To add commands to your new menu, click the Rearrange Commands button. Then locate the name of your new menu in the Choose a Menu or Toolbar to Rearrange list.
5. **Click the Add button and select which features you want to include on your new custom menu.**

## Customizing shortcut menus

Shortcut menus, also called *context menus,* are those little menus that appear when you right-click something. In Office 2003, you can customize these menus in Access, Word, or PowerPoint.

Although thousands of shortcut menus exist, never fear. Here's how to add a new command to a shortcut menu in Access. Follow these steps to add the Help command to the default database background shortcut menu:

1. **Open a database window in Access; then right-click the window to open the default context menu, as shown in Figure 1-4.**

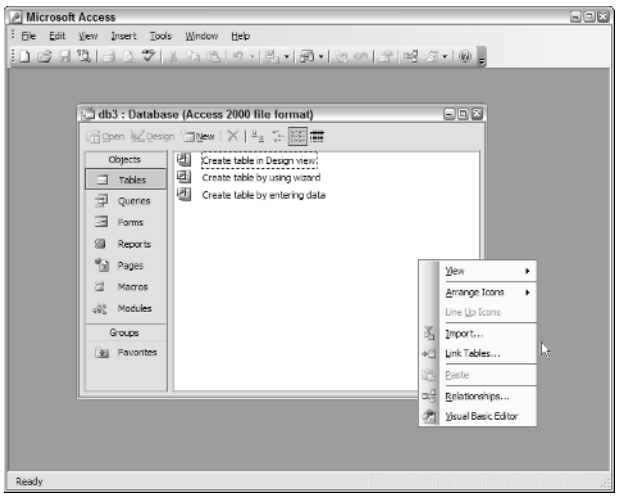


**Figure 1-4:** BEFORE: Many objects in Office 2003 come with a context menu that you can modify.

2. **Choose Tools➪Customize.**
3. **Click the Toolbars tab.**
4. **Select the Shortcut Menus check box.**

   A special shortcut toolbar appears, as you can see in Figure 1-5.

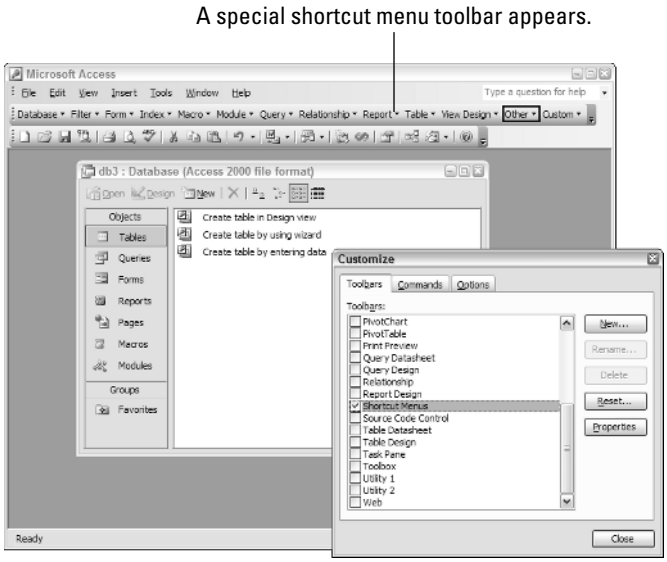


**Figure 1-5:** This special toolbar allows you to choose which shortcut menu to modify.

5. **From this special toolbar, choose Database➪Background, as shown in Figure 1-6.**
6. **Click the Commands tab of the Customize dialog box.**
7. **Choose the category that contains the command you want to add to the shortcut menu.**
8. **Drag the command from the Commands list to the shortcut menu (position it where you want it to appear).**
9. **Drop the command (release the mouse button) into the shortcut menu, as shown in Figure 1-7.**

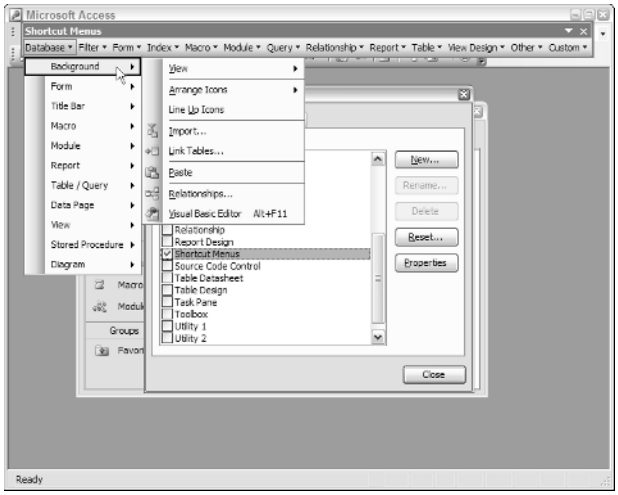


**Figure 1-6:** Choose the shortcut menu you want to modify.

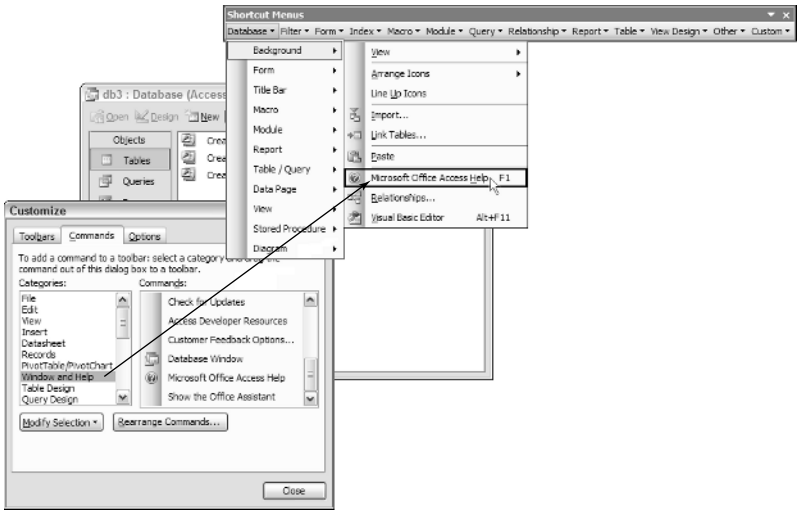


**Figure 1-7:** Drag and drop new commands that you want added to a context menu.

In this example, I add the Help feature to the shortcut menu that pops out when I right-click the background of a database, as shown in Figure 1-8.

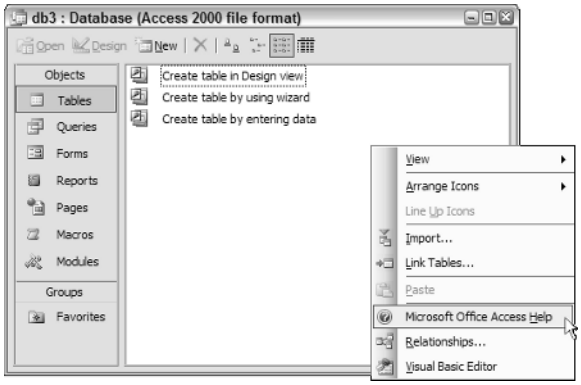


**Figure 1-8:** AFTER: Success! This context menu now has a Help command. Compare this menu with Figure 1-4.


You can edit context menus in many of the same ways you edit ordinary menus — rename, rearrange, add icons, and so on. However, you can't add or delete an entire context menu.

# Personalizing Toolbars

You can manipulate toolbars, tailoring them to suit yourself, much the same way you customize menus. In fact, a toolbar is simply another kind of menu. Although toolbars are more graphic and they are always open, they're just another way for you to trigger behaviors in Office 2003 applications. Some people prefer menus; others consider toolbars more convenient. (You say *toe-may-toe,* and I say *toe-mah-toe.*) As usual in Office, how you work is largely up to you, as long as you know how to modify the applications. After all, it's your work surface, so you should be able to decide where things go and how best to manage it, just as you arrange your desk to suit yourself.


In addition to adding built-in commands (such as File➪Open), to menus and toolbars, you can also add macros. A *macro* is simply a short program, designed to work within and improve the efficiency of the application that hosts it. (See Book I, Chapter 2.) Writing macros allows you to really take control of the elements of an Office application and do with it what you will. You can also add special hyperlinks, such as a link to a worksheet or workbook in Excel or to a Web page.

## Adding hyperlinks

As with menus, you modify toolbars via the Customize dialog box. Just add a custom button, change its image (if you like), and name it. You can even turn the button into a hyperlink. Follow these steps to see how to add a hyperlink to CNN News to the standard Excel toolbar.

First add the custom button:

1. **Choose Tools➪Customize.**
2. **Click the Commands tab in the Customize dialog box.**
3. **Click Macros in the Categories list.**
4. **Drag a custom button from the Commands list and drop it on the Standard Excel toolbar.**

   You don't have to detach the Standard toolbar as it's shown in Figure 1-9.

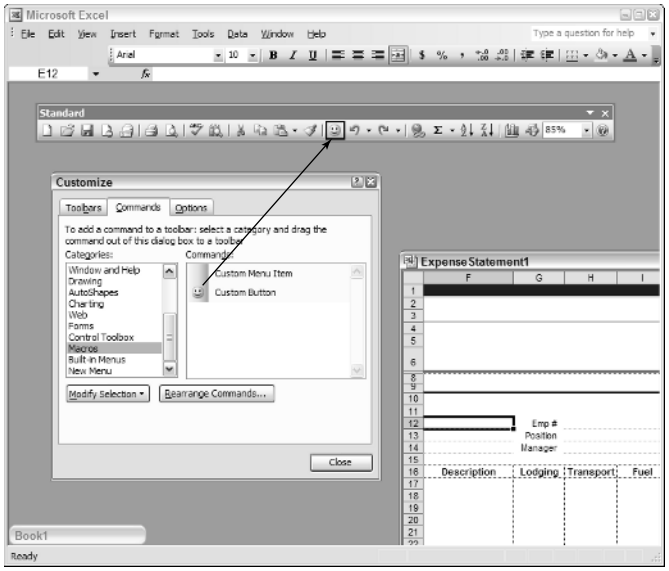


**Figure 1-9:** Add a new button to a toolbar by dragging and dropping.

Don't worry about the smiley face default icon; you can always change this icon by right-clicking it and choosing another graphic. In fact, try that now (with the Custom dialog box still open from the preceding step list).

1. **Right-click the smiley face and choose Change Button Image.**

   You get a palette of images to choose from.

2. **Select the microphone image to remind you that this is CNN, which is broadcast.**

   Smiley is gone, replaced with the microphone image.

Now rename the custom buttom:

1. **Right-click the new button and click the Rename option from the context menu.**
2. **Rename it from Custom Button to CNN.**

Finally, make the button *hot* (into a link):

1. **Right-click the button once again and this time choose Assign Hyperlink, and then Open, from the context menu.**

   The Assign Hyperlink: Open dialog box opens, as shown in Figure 1-10.


**Figure 1-10:** Use this dialog box to create hyperlinks to workbooks, Web sites, and so on.


Note in Figure 1-10 that you can link to various destinations: specific cells or ranges in an Excel workbook; files; e-mail addresses; or new documents, workbooks, Word files, or Notepad TXT files.

2. **Type** http://www.cnn.com **into the Address field (refer to Figure 1-10).**
3. **Click OK.**

   The dialog box closes.

4. **Click the Close button on the Customize dialog box.**

Now try your new hyperlink. Click the microphone icon on the Standard toolbar, and you should see CNN appear, with all the latest shocks, scandals, and scary celebrity agony.


Follow essentially the same steps to add links to menus instead of toolbars. Hyperlinks can also be inserted into Excel workbooks (just right-click a cell and choose Hyperlink from the context menu), Word documents (right-click the document), and so on.

Access, however, is, as usual, the odd stepchild and does things its own, different way. You add hyperlinks to reports, forms, and so on in Access. (It just creates a `Label` control containing the link.) However, the links don't work in Access itself. You must output the report to Excel, HTML, Word, or some other host before the links can actually do their job. As you'll see throughout this book, Access often trods a different path than other Office applications. It appears to exist in a parallel, although similar, universe.

## Vaporizing interface elements programmatically

Throughout this book, you'll find all kinds of programming techniques that you can use to exploit and unify Office 2003 applications. Although you've not yet explored the vast VBA language built into most Office applications, create a useful little macro right now while I'm talking about toolbars. You don't have to understand what's happening in the programming at this point: Monkey-see, monkey-do is just fine at this stage.

Many games and programs have a key you can press that removes all the extraneous, distracting menus, help windows, gauges, and other things from the screen. This frees you up to simply see the essentials. It's similar to choosing View➪Full Screen in Word: All the rulers, scroll bars, menus, and toolbars vanish, and you see the immortal words of the document's writer (you) unadulterated by debris. However, note a couple of problems with Full Screen mode in Word: You lose the scroll bars, and an annoying little bar appears right in the document, which allows you to restore the view to Normal mode.

Here I show you how to write a macro to improve on Word's clean-screen mode. In this macro, I preserve the scroll bar, and you won't need that annoying back-to-Normal-mode bar. Instead, you just use the shortcut key combination Alt+V to toggle Full Screen mode on and off. Simple, clean, and — for those of us who like to type on a blank piece of "paper" without distracting icons all over the edges — a real pleasure to use.

### Programming a macro to hide a toolbar

Here's how to write a macro to hide one or more menus and toolbars so that you can selectively clear the screen any way you choose or toggle between sets of toolbars/menus for different purposes.

Many people have the primary menu (File, Edit, View, and so on), and the Standard and Formatting toolbars visible at all times while using Word. But you use them only now and then. Most of the time, you're just typing. Wouldn't it be nice to have a clean screen in which to type? Figures 1-11 and 1-12 show before and after examples.

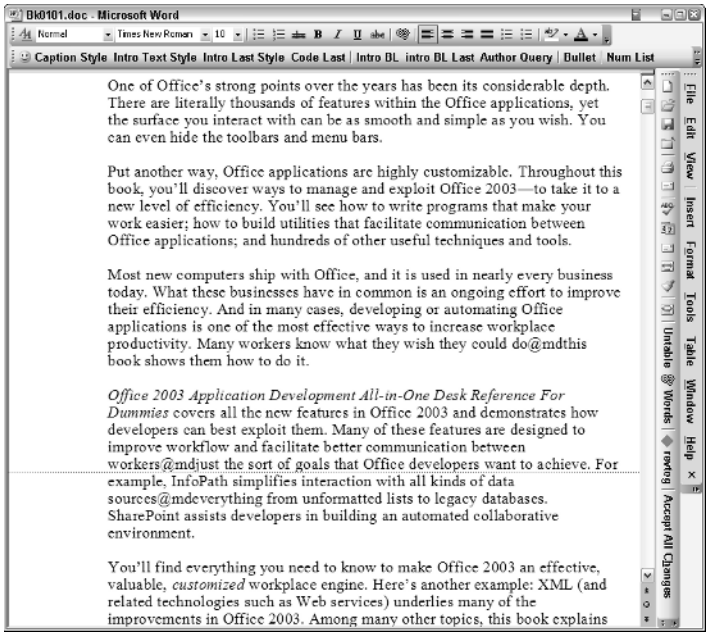


**Figure 1-11:** BEFORE: Toolbars and menus can clutter up a word processor screen.

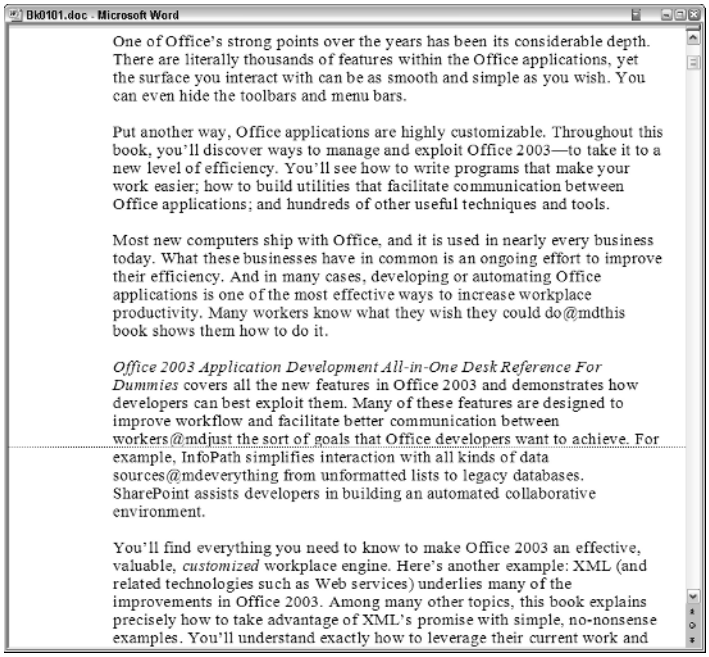


**Figure 1-12:** AFTER: Some people prefer a clean screen to write on.

Here's how to make distracting menus and toolbars disappear, or reappear, every time you press the Alt+V key combination (V for *vanish*).

1. **In Word, choose Tools➪Macro➪Macros.**

2. **Type** Alt+V **in the Name field in the Macros dialog box.**

   You're later going to assign this macro to the Alt+V key combination, so it's useful to name the macro after these keys. It helps you remember.

3. **Click the Create button in the Macros dialog box.**

   A powerful editor opens, about which you can find more in Book II and other places in this book.

   Your insertion cursor (the blinking, vertical line) is now located within your `AltV` macro.

4. **Type in the following VBA commands so that the `AltV` macro (a `Sub`, technically) looks like Listing 1-1.**

**Listing 1-1: AltV Macro**

```
Sub AltV()
'
' AltV Macro
' Macro created 11/25/2003 by Richard
'

If CommandBars("Standard").Visible = True Then

    CommandBars("Standard").Visible = False
    CommandBars("Formatting").Visible = False

Else

    CommandBars("Standard").Visible = True
    CommandBars("Formatting").Visible = True

End If

End Sub
```

5. **Close the Visual Basic editor by clicking the small X icon in the upper-right corner.**

When reviewing the code, you can ignore the lines that begin with single quote marks. They're simply *comments* that the programmer (or in this case, VBA itself) inserted as hints or notes to the programmer. VBA ignores such lines when executing a macro.

You can also ignore these programming commands, but for the curious, the above code translates into English like this: If the Standard toolbar is showing, make it and the Formatting toolbar invisible or else make them both visible. And that's just what you want. Pressing Alt+V toggles their visibility, just the way you toggle a light switch on and off.

### Using macros to remove menus

If you want to hide the main menu, too, you have to do things a bit differently. Menus are not part of a `Menus` collection but are in the `CommandBars` collection. (*Collections* are, simply put, arrays of objects. Like arrays, collections can be manipulated programmatically in loops.) What's more, you can't specify the primary menu (the one with File, Edit, View, and so on) by name but instead must refer to it as the `ActiveMenuBar`. Finally, you can't use the `Visible = False` approach that works with toolbars. Instead, you must use `Enabled = False`. There's no rhyme or reason for these differences: It's just one of the challenges faced by programmers every day. Consistency is attempted in computer languages like VBA but is never fully achieved. Anyway, here's the code that you should add to the above macro to toggle the visibility of the main menu. Insert the bold lines in the places indicated.

```
If CommandBars("Standard").Visible = True Then

    CommandBars("Standard").Visible = False
    CommandBars("Formatting").Visible = False
    CommandBars.ActiveMenuBar.Enabled = False

Else

    CommandBars("Standard").Visible = True
    CommandBars("Formatting").Visible = True
    CommandBars.ActiveMenuBar.Enabled = True

End If
```

Modify this code to add any additional command bars — beyond Standard and Formatting — that you use in Word.

### Assigning the macro to hide menus and toolbars

After you program the macro to hide toolbars and menus (see the preceding sections), all that remains is to assign this macro to the Alt+V key combination. Follow these steps:

1. **Choose Tools➪Customize.**
2. **Click the Keyboard button at the bottom of the Customize dialog box.**
3. **Choose Macros in the Categories list.**

   A new list named Macros appears, with all your macros displayed, including the new one you just wrote, `AltV`.
4. **Choose AltV in the Macros list.**
5. **Click in the Press New Shortcut key field in the Customize Keyboard dialog box.**

   The insertion cursor begins blinking in this field, ready for you to press the key combination that will launch the `AltV` macro.
6. **Press Alt+V.**

   You are informed that Alt+V is unassigned unless you've already assigned it to something previously, in which case you must decide whether to override the previous assignment or choose a new key combination.
7. **Click the Assign button of the Customize Keyboard dialog box.**
8. **Click the Close button of the Customize dialog box.**

## Let Office do the programming

If you're unsure what commands to use when programming in VBA, you can always try a shortcut: Let Office do the programming for you. Here's how. Choose Tools➪Macro➪Record New Macro. The Record New Macro dialog box opens. Click the OK button to begin the recording process. Then do something — type, click the mouse, choose menu options, whatever — while the recorder runs and writes programming for everything you're doing. When you've finished, click the blue square in the Macro Recorder toolbar to stop the recording. Now press Alt+F11 (in Word) to display the VB editor and the programming that was generated for you by the recorder. You can now edit this code, copy and paste it into other macros, or just learn from it. For example, if you choose File➪Save while the recorder is running, you'll find the following code in the VB editor later:

```
Sub Macro6()
'
' Macro6 Macro
' Macro recorded 11/25/2003 by
   Richard
'
    ActiveDocument.Save
End Sub
```

The currently visible Word document is the `ActiveDocument` object. You can append a period after this object's name to perform various tasks that can be accomplished with the `ActiveDocument`.

Now for the fun. Press Alt+V in Word's Normal document view. The toolbars disappear. Press Alt+V again, and they reappear, just as you'd hoped. You can take this technique as far as you want, showing or hiding pretty much whatever you want, whenever, however . . .well, you get the idea. More about VBA in chapters to come.

If you want to go the whole way and create a macro to toggle Full Screen mode, use the following code. (Use this *instead* of the code in the previous sections, not *in addition to.*)

```
ActiveWindow.View.FullScreen = Not ActiveWindow.View.FullScreen
```

## Customizing the Keyboard

Just as you have essentially total freedom to manipulate Office 2003 menus and toolbars, you can also reassign keys to suit your needs. Key combinations can be assigned to trigger all the features in the applications as well as macros and other targets. These combinations are *hot keys* or *shortcut* keys. However, when you open menus via built-in keyboard shortcuts featuring the Alt key (such as Alt+F to open the File menu), this behavior is also called *shortcut keys* or *keyboard shortcuts.* Never mind. Whenever you want, you can change the classic key assignments to whatever you want.


To see the shortcut keys assigned to toolbar buttons when you pause your mouse cursor over a button, choose Tools➪Customize and then click the Options tab. Mark the Show ScreenTips on Toolbars and the Show Shortcut Keys in ScreenTips check boxes to select them. Making these changes affects the behavior of all the other Office 2003 applications. (***Note:*** Excel displays only the Show ScreenTips on Toolbars check box, so you can't make this change from Excel.)

## Restoring Classic Key Behaviors

When you first start using Office 2003 Word, you might notice that several traditional keyboard behaviors have been rather strangely altered. For example, the Delete key has for decades been used to delete a selected block of text. Now, when you select (drag) some text (so it reverses color, to white on black), pressing the Delete key merely displays a little (and for most of us who are capable typists, highly annoying) question: Delete block? No (Yes). Repeatedly pressing the Delete key has no effect. You must also press Y to actually perform the job that pressing the Delete key used to accomplish.

Similarly, keys that maneuvered you through a document have been reassigned. You used to get to the start of a line of text by pressing the Home key. Now you must press Home+←. You used to be able to press Ctrl+Home to get to the beginning of the document; now this displays the Find and Replace dialog box, with the Go To tab selected.

If you're finding these strange behaviors, you won't be able to remap these keys by using the usual approach (choosing Tools➪Customize and then clicking the Keyboard button). The Delete key is set to Edit➪Clear, which is what it's supposed to say. It just behaves oddly.

Here is the solution: For reasons unknown, when Word is installed, it sometimes switches on the Navigation Keys for WordPerfect Users option. To fix this and restore your familiar Word behaviors, choose Tools➪Options and then click the General tab. Deselect the Navigation Keys for WordPerfect Users option.

## Getting Online Help

Microsoft provides extensive online help for developers and programmers, and its Office information is no exception.

Your portal to Office 2003 help online is `http://office.microsoft.com`. (You'll find a link to this site in the Office applications' Help menu.) At this location. you find a list of the individual Office applications and utilities. (See the left side of Figure 1-13.)

Click the Access link, for example, and you'll be told that in Office 2003, you need to upgrade to Jet 4.0 if you want Access to be able to offer all its features yet at the same time block unsafe expressions that could cause virus-like damage.

Another useful online resource is MSDN, the Microsoft Developer Network. Here at `http://msdn.microsoft.com`, you can find advanced tutorials, a search engine, white papers, downloads, free software trials, and other often useful items. MSDN is also a subscription service that sends out early versions of Microsoft products, CDs full of various kinds of programmer-oriented tools, and so on. However, you don't have to be a subscriber to take advantage of the wealth of information online.

If you're looking for answers to specific questions, try joining one of the newsgroups dedicated to the various Office applications. Try this address: `http://support.microsoft.com/newsgroups/default.aspx`. Then drill down until you find the application, topic, and messages of interest to you.


**Figure 1-13:** This is the gateway to all kinds of information at Microsoft's Office site.