

Chapter 1: Introducing Access 2003

In This Chapter

- ✓ Getting a handle on Microsoft Access
- ✓ Listing the eight types of Access objects
- ✓ Laying out some essential database concepts

Access is the Microsoft database management program, part of the Microsoft Office suite of programs. A database management program enables you to maintain *databases* — collections of data arranged according to a fixed structure. The structure makes the information easy to select, sort, display, and print in a variety of formats. With Access, you can create and maintain as many databases as you need — you can even share them with other people over a local area network or the Internet.

Access is a general-purpose program that works with almost any kind of information. A database can be as simple as a list of addresses to replace your card file. Or you can create a wine cellar database with information about each bottle in your cellar or a bookstore inventory database with information about books, publishers, customers, and special orders. Access can also handle complex databases that contain lots of types of information and lots of customized programming.

An Access database can contain lists of records about almost anything, from sales to sports scores. Unlike a spreadsheet program, Access makes displaying your information in lots of different formats easy, including alphabetical listings, formatted reports, mailing labels, and fill-in-the-blank forms.

Access 2003 comes with Microsoft Office 11. Previous versions of Access have also been part of the Office suite — Access 2002 in Office XP, Access 2000 in Office 2000, and Access 97 in Office 97. Because Access is part of Microsoft Office, sharing information with Word documents and Excel spreadsheets is easy. This book describes Access 2003, but except where noted, everything we say is true for Access 2000 and 2002, too.

Access includes a powerful programming language (VBA, described in the section, “Modules for writing your own programs,” later in this chapter).

12 *The Eight Types of Access Objects*

And you can use Access to work with data stored in large corporate databases. These features make Access a great choice for creating *front ends* for corporate databases — which means you can use Access to edit, display, and print data from larger, shared databases, without having to do any programming.

The Eight Types of Access Objects

Access databases are made up of *objects* — things you can create, edit, and delete, with names and settings. *Object-oriented* systems allow you to create things one piece at a time, using pieces that fit together.

Access contains various kinds of objects, including objects for storing, displaying, and printing your data, as well as objects that contain programs that you write. At first, you'll probably use only a few types of objects, but as you customize your database, you may end up using them all. You start with *tables* for storing data, *forms* for editing data on-screen, *reports* for printing data, and *queries* for selecting and combining data. Later, you may create *macros* and *modules*, which contain programs that you write; *projects*, which work with corporate data; and *data access pages*, which allow you to edit Access data via a Web browser.

In this section, we cover each of the main types of Access objects: tables, queries, forms, reports, macros, modules, projects, and data access pages.

Tables for storing your data

Tables are where you put your data. A table is an Access object that is made up of a series of *records*, which are like the index cards that make up an address list. Each record contains information in the same format. In an address list, each record contains information about one person: name, address, and other facts. Each individual piece of information — such as the first name, the last name, or the street address — is called a *field*.

Your database can contain many tables. A bookstore database can contain a table of books (with title, publisher, price, and other information about each book), a table of vendors from whom you buy books (with company name, address, discount terms, and other information about each vendor), and maybe a table of your regular customers (with name, address, and other information). Figure 1-1 shows a table of names and addresses. Each row is a record, and the fields are shown in columns.

After you set up tables in your database and type in (or import) information, you can sort the records, select records that match a criterion, and then display and print the records.

Figure 1-1:
A table
contains
a list of
records,
and each
record is
made up
of fields.

Address Book : Table					
ContactID	First Name	Last Name	Company	Address1	
9	Wilma	Wannabe	Wannabe Whistles	1121 River Road	Su
10	Frankly	Unctuous		734 N. Rainbow Dr.	
11	Tori	Pines	Arbor Classics	345 Pacific Coast Hwy.	Su
12	Mary	Smith	Grandview Middle School	7724 Prospect St.	
139	Margaret	Angstrom		P.O. Box 1295	
140	Simpson	Sarah		1370 Washington Lane	
141	Nancy	O'Hara	ABC Productions	Haverston Square	113
142	Hortense	Higglebottom		P.O. Box 1014	112
143	Penny	Lopez		P.O. Box 10	
144	Bill	Gates	Microsoft	3075 112th Avenue NE	
145	Hank & Matilda	Starbuck		323 Shire Lane	
146	Scott and Natalie	Schumack		228 Hollywood Drive	
147	Linda	Peterson		823 Paseo Cancun	

Book I
Chapter 1

Introducing
Access 2003

Proper design of your tables — choosing how many tables to create and which fields are stored in which table — is key to creating a usable and flexible database. Chapter 3 of this book includes a step-by-step procedure for designing your database, and Book II explains how to create tables and fill them with data.

Queries for selecting your data

Queries are a means to slice and dice your data. The most commonly used type of query helps you select data from a table, perhaps to select which records you want to include in a report. You can create a query that shows you all the people in your address book who live in Vermont, or all those for whom you don't have a phone number. When you create this type of query, you enter *criteria* that specify what values you want to match in specific fields in the tables (for example, VT in the State field, or nothing in the Phone Number field, or both).

Is a spreadsheet a database?

Many people use spreadsheet programs, such as Microsoft Excel or Lotus 1-2-3, to store lists of records. Some spreadsheet programs have limited database capabilities, but they aren't designed to do as much as a database program. You can use a spreadsheet to store an address list, and you can enter, edit, delete, and sort the addresses (one per row on the spreadsheet),

but printing mailing labels or form letters is a major chore. Spreadsheets do not think of your data in terms of tables, records, and fields, but rather in terms of cells (the basic unit of a spreadsheet) in rows and columns. We bet you're glad you're using Access for your database work!

14 The Eight Types of Access Objects

You can also use queries to combine information from several tables. A bookstore database may store book author names in the Books table and book ordering information in the Purchase Orders table. A query can pull information from both of these tables — you can create a query to show all the Terry Pratchett novels that you ordered for the last month, for example. Queries can also create calculated fields, including totals, counts, and averages.

Another type of query is the *action query*, a query that does something to the records that you select. You can create action queries to copy records from one table to another, to make a change to all the records you select, or to delete records you select. *Crosstab queries* help you analyze the information in your tables, by summarizing how many records contain specific combinations of values.



Queries are the way that you get useful information out of your tables, and you'll probably create zillions of them as you play with your database. Book III explains how to create and use queries of all kinds.

Forms for editing and displaying your data

An easy way to do data entry, especially if you enter data into more than one related table, is to use a *form*. A form displays the information from one or more tables on-screen. You can edit the data or type in new records. You can choose the layout of the table's information on the form — you can specify the order in which the items appear, you can group items together with lines and boxes, and you can use pull-down lists, radio buttons, and other types of on-screen controls for entering and editing data. The form in Figure 1-2 shows information from the Product table of an online store.

Figure 1-2:
A form
shows
information
stored in
one or more
tables, for
data entry
and editing.

You can build intelligence into forms, too — you can program boxes that automatically capitalize what you type in, or that check your entry against a table of valid values.



After your database goes into production — you use it for its intended application — forms are the most-used Access object. Book IV explains how to design, create, modify, and use forms.

Reports for printing your data

Forms are primarily designed to appear on-screen; *reports* are designed for printing, as shown in Figure 1-3. Like forms, reports display information from tables, and you can choose the layout of the information. Most reports are based on queries; you use a query to choose the information to appear in the report. The report design defines the order in which records appear, which fields appear where, fonts, font sizes, lines, and spacing.

Figure 1-3:
A report displays information from one or more tables, for printing.

Directory	
Angstrom, Margaret P.O. Box 1295 Daneville, CA 92067 margaret@angstrom.com Date Entered 1/10/2003	Contact ID: 139 (Customer) Phone (713) 555-3232 Fax (713) 555-5403 <input type="checkbox"/> Tax Exempt Tax Exempt ID
Bean & Beluca 2526 E. 36th St. N. Cir Wichita, KS 67219 purchasing@beanbeluca.com Date Entered 1/10/2003	Contact ID: 169 (Vendor) Phone (358) 555-8215 Fax (358) 555-0250 www.beanbeluca.com <input type="checkbox"/> Tax Exempt Tax Exempt ID

In addition to reports on normal paper, you can create reports for printing on envelopes, labels, or other printed forms. Access comes with report wizards that make creating fancy reports easy. It can also print charts and cross-tabulations (*crosstabs*) based on the data in your database.



Book V covers how to create and print reports, charts, and crosstabs.

Macros for saving keystrokes

Access includes two separate programming languages: macros and VBA. *Macros* are programs that automate commands you give when you use Access. For example, you can write a macro that moves the cursor to the

16 *The Eight Types of Access Objects*

last record in the Orders table whenever you open the Order Entry form. (What are the chances that you want to edit your very first order? More likely, you want to edit the last order or enter a new order.) Or you can write a macro that moves your cursor to the next applicable blank in a form, based on the entries you made so far.

After you know how to create macros, you can create buttons on your forms that run the macros with a quick click. You can also tell your form to run a macro automatically whenever you move to a field on the form, or enter data into the field — handy!



You don't have to be a programmer to create macros. Access helps you write them by providing menus of commands. Book VI explains how to create nifty and useful macros to clean up your data entry and a number of items automatically.

Modules for writing your own programs

Okay, now we come to the serious programming stuff: *modules* — another term for VBA programs. *VBA* (Visual Basic for Applications) is a programming language based on the age-old BASIC language. Macros are fine for saving a few keystrokes or cleaning up the data you enter in a field, but when the going gets complex, you can use VBA.

Say you receive orders for your online store via e-mail messages in a specific format. You can write a VBA program to read these e-mail messages from your e-mail mailbox and create records in your Orders table. Why cut and paste when a VBA program can do the job? Or you could write a VBA program to be called when you click the Ship This Order button on your Orders form: The program prints a packing slip, prints a mailing label, updates your inventory numbers, and generates an e-mail message to the customer. Why should you remember to do all those tasks when a VBA program can do them all?



Programming isn't for the faint of heart, but when you have the rest of your database up and running, take a look at Book VIII for an introduction to VBA programming. Writing small programs is not hard, and after you get used to programming, who knows what you'll end up creating!

Projects for accessing shared databases

If you're not a corporate user (or part of some large organization), skip this one — you'll never use the special databases called *projects*. Still here? Fine. An Access project is a special type of database in which the actual data you normally store in your Access tables goes to a large corporate database, to

be stored there instead. Your Access project file contains only your tools — the forms, reports, pages, macros, and VBA modules you need for creating or modifying projects. You connect to the corporate database to get your data from the corporate database's tables and queries.

The great thing about Access projects is that they give you a consistent way to use Access effectively when you work with corporate data. Instead of having to learn the ins and outs of database programming with some large, scary program, you can use Access forms and pages to look at the data and reports to print it out.



Book IX, Chapter 2 walks you through creating an Access project. You need to get information about your organization's database from your Information Services department, including a password to give you permission to use the files.

Pages for editing data in your Web browser

Forms are great for entering and editing data, but you have to be running Access to use them. What if you want non-Access folk to work with the information in your database? *Pages* (or *data access pages*) enable you to create Web pages containing forms, so anyone with the Internet Explorer browser can edit records in the tables in your database. Anyone, that is, with permission to do so — you need to set up the security features to control who can look at and change your information.



Book IX, Chapter 1 talks about pages and how to make them. After you create pages for use with your database, you also find out how to upload them to your Web server so that other people can use the pages. Book VII, Chapter 3 describes the Access security features to control who has permission to see or change each field in each table.

Essential Database Concepts

Here are the Four Commandments of databases. (Aren't you relieved there aren't 10?). You'll find lots more important rules and guidelines throughout this book as you discover how to work with various Access objects, but here are some that apply right from the start, no matter what kind of database you are using.

- ◆ **Store information where it belongs, not where it appears.** Where you store information has nothing to do with where it appears. In a database, you store information in tables based on the structure of the information. (Don't worry — Chapter 3 of this book explains how to figure out the

18 *Essential Database Concepts*

structure of your data.) Each piece of information likely appears in lots of different places. For example, in a database for an online bookstore, book titles and authors' names appear on your invoices, purchase orders, and sales receipts. But the right place to *store* those book titles and author names is in the Books table, not in the Sales table or the Purchase Orders table.

- ◆ **Garbage in, garbage out (GIGO).** If you don't bother to create a good, sensible design for your database, and if you aren't careful to enter correct, clean data, your database ends up full of garbage. A well-designed database is easier to maintain than a badly designed one, because each piece of information is stored only once, in a clearly named field in a clearly named table, with the proper validation rules in place. Yes, it sounds like a lot of work, but cleaning up a database of 10,000 incorrect records is (pardon the understatement) even more work. See Book II, Chapter 5 for ways to avoid GIGO.
- ◆ **Separate your data from your programs.** If you create a database to be shared with or distributed to other people, store all the tables in one database (the *back end*) and all the other objects in another database (the *front end*). Then you link these two databases together to make everything work. Separating the tables from everything else makes updating queries, forms, reports, or other stuff later without disturbing the data in the tables easy. (See Book VII, Chapter 1 for how to separate a database into a front end and back end.)
- ◆ **Back up early and often.** Make a backup of your database every day. With luck, your office already has a system of regular (probably nightly) backups that includes your database. If not, make a backup copy of your database at regular intervals, and certainly before making any major changes. (See Book VII, Chapter 1 for how to make backups.)