

Creating Access Tables

In order to manage data with Access, you must store the data in tables. If the data you want to work with is already stored in a computer database somewhere, you probably can use Access to get at it as Chapter 7 explains. But if the data exists on paper only or it isn't yet available or it's in a format that Access can't import or link, your first step is to structure tables that will store the data. This chapter is all about creating tables.

If you used the Database Wizard to create a database, you can skip this chapter and continue with Chapter 8 to learn more about entering data into your tables. You can always return to this chapter later if you need to add, change, or delete table fields.

If you'd like to review or dive more deeply into topics covered in this chapter, go to the Access Contents Index; open the *Creating, Importing, and Linking Tables* book; and then explore the subtopics. For hands-on practice with creating and opening a database and its tables, see Chapter 3.

Creating the Database

If you haven't done so already, you must create a database in which to store your tables as explained in Chapter 5. If you have created a database already, be sure to open that database (see Chapter 1).

**NOTE**

Remember, a table is not the same as a database. A database can contain any number of tables. So don't create a new database each time you want to create a table. As long as your new tables are related to other tables in the current database in some way, you should continue to add those new tables to the current database.

Using the Table Wizard to Create Tables

Want a table in a hurry? Then follow these steps for using the Access Table Wizard:



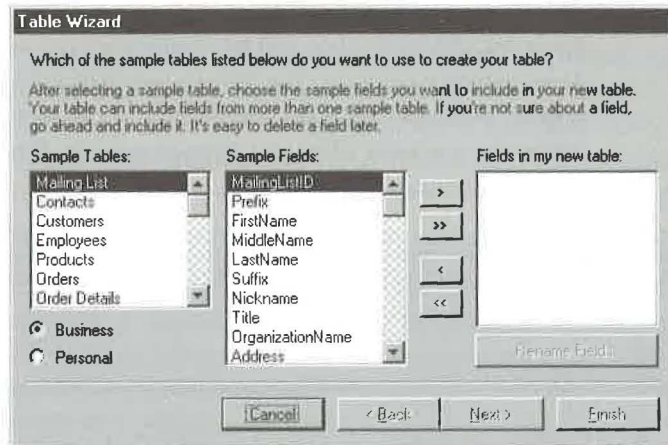
1. Click on the Tables tab in the database window and then click on the New button. Or click on the drop-down arrow on the New Object toolbar button (shown at left) and choose New Table. Or choose Insert ► Table. You'll see this New Table dialog box next:



2. Double-click on Table Wizard to open the Table Wizard dialog box, shown in Figure 6.1.
3. Choose either Business or Personal from the option buttons.
4. Scroll through the list of tables and then click on the sample table name that best describes the information you want to store (Mailing List, Contacts, Customers, or whatever).
5. Click on a field name in the Sample Fields list for each field you want to include in the table and then click on the > button (or double-click on a field name). Or click on the >> button to copy all the sample fields to your table. The field names will be copied to the Fields in my new table list in the order you specify. Here are some ways to manage the Fields in my new table list:
 - **To delete a field**, click on that field in the Fields in my new table list and then click on the < button.
 - **To delete all fields in the Fields in my new table list**, click on the << button.

FIGURE 6.1

The Table Wizard dialog box will help you create a table.



- To move a field up or down in the list, first delete that field from the Fields in my new table list. Next, in the Fields in my new table list, click on the place where the field should appear. Finally, double-click on the appropriate field name in the Sample Fields list.
- To change a field's name, click on that field in the Fields in my new table list, click on the Rename Field button, type a new name for the field, and then click on OK.



As for all the Wizards, you can click on the Back button to back up to a previous dialog box, Next to continue to the next dialog box, Finish to zip to the finish line, and Cancel to exit the Wizard without creating the table. Always look at each Wizard dialog box carefully for previews, tips, and other information that will help you decide what choices to make and what to do next.

6. Click on the Next button when the Fields in my new table list contains the fields you want to include in your table. The next dialog box suggests a name for the new table.
7. Leave the suggested name unchanged (if a table with this name doesn't already exist in the database) or type a new name. See the sidebar "Object Naming Rules" for details about how to name tables and other objects.
8. Assign a primary key. You have two options:
 - To have Access make the decision for you (the easiest method), select Yes, Set A Primary Key For Me, and click on Next.

- **To make your own decisions about the primary key**, select No, I'll Set The Primary Key, and click on Next. In the next dialog box, choose from the drop-down list whichever field will be unique for each record. You'll also need to tell Access which type of data the primary key field will contain (numbers that Access assigns, numbers that you enter when you add new records, or numbers and/or letters that you enter when you add new records). When you're done choosing primary key options, click on Next.

**NOTE**

If you're not sure about how to handle primary key fields just now, don't worry about it. Select Yes, Set A Primary Key For Me in step 8 and click on Next. You'll learn more about primary keys later in this chapter, under "Setting a Primary Key."

9. If your database already contains at least one table, you'll be asked to specify whether your new table is related to other tables in your database. Here's what you can do:
 - **If you're not sure about the table relationships**, click on Next to continue with the next dialog box for now. You can define relationships later, if necessary, when you understand more about them. "Defining Relationships Among Tables," later in the chapter, explains relationships and shows how to define them anytime.
 - **If you do know which relationships you want to define** between the table you're creating and another table that already exists in the database, click on the related table in the list and then click on the Relationships button. In the next dialog box, tell Access how your new table is related to the existing table and click on OK. Repeat this step until you've defined all the relationships you want and then click on Next.
10. Follow the prompts in the final dialog box (which displays a checkered flag) to tell the Wizard what to do after creating the table. Then click on the Finish button to create the table.

You'll be taken to the table design window or to the datasheet view or to a form, depending on your choice in step 10. To return to the database window now, choose File ► Close from the Access menu bar or click on the Close button on the window that appears or press Ctrl+W. If you're prompted to save your changes, click on Yes or No as appropriate.

NOTE

When you use the Table Wizard to create a table, Access automatically sets up input masks for certain fields (such as those that store telephone and fax numbers, postal codes, and dates). *Input masks* make data entry easier by controlling where data is entered, what kind of data is allowed, and how many characters you can enter. You'll find more about input masks later in this chapter.

Object Naming Rules

Access gives you considerable freedom when assigning names to tables, table fields, forms, reports, queries, macros, and modules. Still it will complain if you don't follow the naming rules, which are as follows:

- The name cannot be the same as any other object of the same type within the database. For example, you can't have two tables named MailingList; however, it's OK to have one table named MailingList1 and another named MailingList2.
- You cannot give a table and a query the same name.
- When naming a field, control, or object, make sure your name isn't the same as the name of a property or other element that Access is already using. This rule is especially important if you're writing Visual Basic code.
- The name can be up to 64 characters, including spaces. However, the name cannot start with a space.
- The name can include any combination of letters, numbers, and spaces.
- The name can include punctuation characters *except* a period (.), an exclamation mark (!), an accent grave (À), or brackets ([]).
- If you plan to write programs for use with your database, avoid using spaces in object names. For example, use *CustomerInfo* instead of *Customer Info* (for a table name) or *LastName* instead of *Last Name* (for a field name). Programming is easier when the object names do not include spaces.
- The name cannot include control characters (ASCII 00 to ASCII 31).
- When naming fields, avoid choosing names that are the same as built-in Access function names or property names. Strange things may happen if you do.

For more about naming rules, look up *Naming Rules* in the Access Answer Wizard and then double-click on appropriate sub-topics. For a complete list of Microsoft Access specifications, look up *Specifications* in the Access Answer Wizard and then double-click on *Look Up Microsoft Access Specifications* under *How Do I*.

We'll talk about the table design window and ways to change a table's structure later in this chapter. But first, let's look at ways to create a table without a Wizard. The process begins with planning a table from scratch.

Planning a Table from Scratch

Instead of using the Table Wizard to create a table, you can plan and create a table from scratch. Your first step is to decide which fields to include in your table. If you're working from a fill-in-the-blank paper form, this task can be easy. Generally speaking, you can create one field for every "blank" on the form. An accountant, for example, might create a table that has a field for each blank on a Federal 1040 tax form. There's one catch to this one-blank, one-field approach: You generally should not include fields that contain the results of calculations. Why not? Read on.

Do Not Include Calculated Fields

Access can perform instant, accurate calculations on any data in your table and then display the results in queries, forms, and reports (which you'll learn about in upcoming chapters). So, for best results, do not create fields that store the results of a calculation. Here are the potential problems with storing calculation results in the table:

- You risk printing faulty totals or results that aren't based on up-to-the-minute data.
- You're wasting disk space. After all, there's no need to store what Access can calculate on-the-fly for you.
- You'll need to do the calculations yourself. Unless you're some kind of human calculator, doing the calculations yourself is a waste of time and can lead to errors.

In short, tables should contain raw data only—just the numbers you'll need to base calculations on later—and not the results of any calculations.

Creating a Table without Using a Wizard

Follow these steps to create the table from scratch, without using the Table Wizard:

1. Click on the Tables tab on the database window and then click on the New button. Or click on the drop-down arrow on the New Object toolbar button and choose New Table. Or choose Insert ► Table.
2. Double-click on Design View in the New Table dialog box that appears next.

Defining a Table's Fields

Next you must define the fields in your table by following the steps below. As you define the fields, glance at the hint box in the table design window for guidance. (You also can press F1 for more information as you follow each step.)

1. Type a field name (up to 64 characters including blank spaces) in the Field Name column. Field names must follow the rules given earlier in this chapter (see the sidebar "Object Naming Rules").
2. Click on the Data Type column next to the field name and select the appropriate data type from the drop-down list. (See "Choosing Appropriate Data Types" below for more information.)
3. Click on the Description column and type a description of the field (a description is optional, but very helpful). This description will appear on the status bar later when you're entering data.
4. (Optional) Click on the General or Lookup tab in the Field Properties area and set properties for the field you're defining. See "Setting Field Properties" later in this chapter.
5. Repeat steps 1 through 4, putting each field definition on its own row until you've defined all the fields in your table.

When you've finished defining the table's fields, save the table structure as discussed shortly under "Saving a Table Structure."



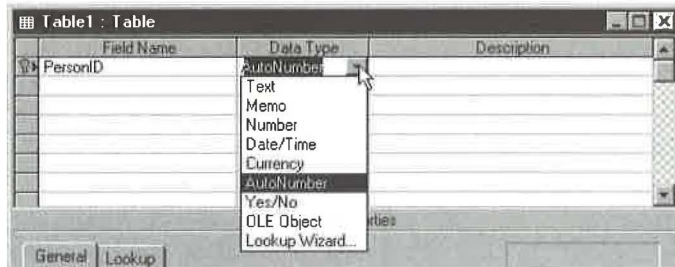
If you prefer to let a Wizard guide you through setting up a field, click in the Field Name column where you want the new field to appear. Then right-click and choose Build from the shortcut menu or click on the Build toolbar button (refer to Figure 6.2). The Field Builder will guide you through the remaining steps.

Choosing Appropriate Data Types

Access can store different types of information in different formats. So when you're defining a table without a Wizard, you must think about what type of information will be stored in each field.

To define the data type of a field, click in the Data Type column next to the field name and then click on the drop-down list button that appears. You'll see the list

shown below and you can select a data type by clicking on it. Table 6.1 summarizes the data types you can choose.



TIP Be sure to assign the Text data type—rather than the Number data type—to fields such as telephone numbers, fax numbers, postal codes, and e-mail addresses. Unlike the Number data type, the Text data type lets you enter punctuation characters and letters, as in these examples: (520)555-5947 for a telephone or fax number; 73444,2330 or Hanley@RNAA47.com for an e-mail address; 85711-1234 or H3A3G2 for postal codes.

TABLE 6.1: ACCESS DATA TYPES

DATA TYPE	STORES
AutoNumber	A number that's assigned automatically and never changes again. <i>Tip:</i> Use the New Values property on the General tab to control whether numbers are assigned incrementally or randomly.
Currency	Dollar amounts.
Date/Time	Dates (e.g., 12/31/96) and times. <i>Tip:</i> Use the Format property on the General tab to control the appearance of the date.
Hyperlink	Hyperlink addresses.
Lookup	Values that come from another table, a query, or a list of values you supply. <i>Tip:</i> Use the Lookup tab properties to define a lookup field or choose the Lookup Wizard data type to set up the lookup field automatically.
Memo	Large bodies of text up to 64,000 characters in length.

Continued

TABLE 6.1: ACCESS DATA TYPES (CONTINUED)

DATA TYPE	STORES
Number	True numbers such as quantities. <i>Tip:</i> Use the Field Size, Format, and Decimal Places field properties on the General tab to control the size and appearance of numbers.
OLE Object	Any OLE object such as a picture, sound, or word processing document.
Text	Any written text up to 255 characters in length, numbers that you won't be using in arithmetic calculations, and certain numbers like codes such as zip codes, phone numbers, or product codes that contain letters, hyphens, or other nonnumeric characters.
Yes/No	A True or False value only. <i>Tip:</i> Use the Format property on the General tab and the Display Control property on the Lookup tab to control the appearance of the field's contents.

To learn more about how to decide what data type to use for fields in a table, click in the Data Type column on the table design window and press F1. Or search for *Data Types* topics and subtopics in the Access Help Index.

Defining Field Properties

You can change a field's properties (characteristics) using options on the General and Lookup tabs below the Field Properties area in the table design window's lower pane. Different data types offer different properties.



WARNING

The Field Properties area of the table design window shows the field properties for only one field at a time. Check to make sure the ► symbol on the row selector is pointing to the appropriate field name in the upper pane before you change properties in the lower pane.

To set a property for a field, follow these steps:

1. Select the appropriate field in the table design window's upper pane.
2. Click on the appropriate tab (General or Lookup) in the Field Properties area in the table design window's lower pane.
3. Click in the box next to the property you want to set.

4. Do any of the following:

- Type a value for the property.
- Click on the drop-down arrow (if one appears) next to the property and click on an option in the list that appears. For properties that offer a drop-down list, you also can double-click on the appropriate property box to cycle through the available values for that property.
- Get help in setting the property by clicking on the Build button (if one appears) or right-clicking on the field and choose Build.

You can get immediate help as you define field properties by pressing F1. You also can search for *Fields, Properties* in the Access Help Index.

Important General Field Properties

Here (in alphabetical order) are the most important field properties on the General tab. (We'll get to the Lookup properties later in this chapter.)

Allow Zero Length If Yes, the field will accept an "empty string" as a valid entry, even if the Required property is set to Yes. That empty string will appear as two quotation marks with nothing in between ("") when first typed into the field; those quotation marks will disappear when you move the cursor to another field.

Caption Lets you define an alternative name for the field to be used in datasheet view and as labels when you create forms and reports. The caption offers a handy way to make your datasheet columns and labels more readable when field names do not contain spaces. *Example:* If you've named a field *LastName*, specify that field's Caption property as the more readable text *Last Name*.

Decimal Places Lets you specify the number of digits to the right of the decimal separator in a numeric field. Choose "Auto" to have the Format property determine the number of decimal places automatically.

Default Value Lets you define a value that's automatically inserted into the field; you can type a different value during data entry, if necessary. (See Table 6.2 for examples.) The default value for a Text field is the empty string; for a Number or Currency field, it's 0.

Field Size Lets you specify the maximum length of text allowed into the field or the acceptable range of numbers. The default size for text is 50 and for numbers is Long Integer, although you can change these settings by choosing Tools ► Options, clicking on the Tables/Queries tab, and changing values in the boxes under Default Field Sizes (see Chapter 15).

Format Lets you define the appearance of data in the field.

NOTE

Access *doesn't* pad text that's shorter than its allotted width to fill out the rest of the field. Hence there's no disk consumption penalty for making the size of a Text field wider than it needs to be. However, a smaller maximum field size can conserve memory and speed up processing.

TABLE 6.2: EXAMPLES OF DEFAULT VALUES YOU CAN ASSIGN TO A FIELD

DEFAULT VALUE	FILLS FIELD WITH
=Date()	Today's date (use the Date/Time data type)*
=Now()	Current date and time (use the Date/Time data type)*
0	The number zero (use with Number and Currency data types)
Yes	A "True" setting (use with Yes/No data type)
No	A "False" setting (use with Yes/No data type)
CA	The letters CA (for a Text field that defines the two-letter state abbreviation for California, used in the United States)

* Use the Format property on the General tab to determine the appearance of the field's contents

Indexed Lets you choose whether to index this field and whether to allow duplicates in the index. See "Defining Indexes" later in this chapter for more details.

Input Mask Lets you define a pattern for entering data into the field. For help with creating the mask for a text or date/time field, click on the Build button after selecting this property. The Input Mask Wizard will guide you through each step. You also can press F1 for help when the cursor is in the Input Mask property box.

Required If set to Yes, the field cannot be left blank.

Validation Rule Lets you create an *expression* that tests data as it comes into the field and rejects faulty entries. (See Table 6.3 for examples.)

TIP

In addition to specifying a validation rule, you can limit the entry in a field to values from another table. To do so, you can (1) define relationships between tables; (2) define lookup fields; or (3) create a drop-down list in a form. We'll explain the first two methods later in this chapter and the third method in Chapter 13.

Validation Text Defines the error message that will appear on the screen when faulty data is entered into the field. When writing the validation text, it's best to indicate which field is invalid so the user can more easily understand what's wrong during data entry. (See Table 6.3 for examples.)

**TIP**

To test all your validation rules against existing data in the table, choose Edit ► Test Validation Rules ► Yes.

TABLE 6.3: EXAMPLES OF VALIDATION RULE EXPRESSIONS AND TEXT FOR ERRORS

VALIDATION RULE	POSSIBLE VALIDATION TEXT	HOW IT LIMITS ENTRY
>0	The Unit Price must be greater than 0.	Disallows 0 or negative number in a Number or Currency field.
<>0	A Rating of 0 is not acceptable.	Allows any negative or positive number, but not zero.
Between 1 and 100	The Rating must be between 1 and 100.	Accepts only numbers in the range of 1 to 100 (inclusive) in a Number or Currency field.
>=Date()	Sorry, no Order Date backdating is allowed!	Allows only today's date or later dates in a Date/Time field.
>=#1/1/96#	Sorry, First Payment date must be January 1996 or later.	Prevents dates earlier than January 1, 1996, in a Date/Time field.

Setting a Primary Key

A *primary key* is a field (or fields) that uniquely identifies each record, much as a license plate uniquely identifies each car on the road. When you define a primary key, you tell Access to do three things:

- Make sure no two records in the table have the same value in the field (or fields) that define the primary key
- Keep records sorted (ordered) by the entries in the primary key field

- Speed up processing

A primary key can be just one field, or it can consist of two or more fields. When two or more fields define a primary key, Access doesn't consider records to be duplicates unless the combined contents of all the fields in the primary key are identical.

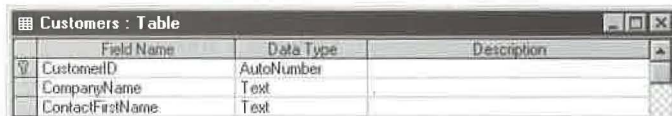
NOTE

No field that is part of a primary key can ever be left blank during data entry.

To set a primary key in your table design, do the following:

1. Select the field you want to use as a primary key by clicking on the row selector button to the left of the field name. Or if you want to select multiple fields, hold down the Ctrl key and click on the row selector for each field you want to define.
2. Click on the Primary Key toolbar button (shown at left). Or choose Edit ► Primary Key from the menu bar. Or right-click on the highlighted row selector and choose Primary Key.

The field(s) you've set as the primary key will have a key icon in the row selector as shown below.



	Field Name	Data Type	Description
☑	CustomerID	AutoNumber	
	CompanyName	Text	
	ContactFirstName	Text	

If you change your mind about assigning a field as a primary key, just repeat the two steps above.

Defining Indexes

You can add an index to a field as a way to speed up sorting and searching on that field. Not all data types can be indexed, but any field that has a property named Indexed at the bottom of the General field properties tab is a candidate for indexing. Each table in your database can have up to 32 indexes.

Be aware that indexes can slow down data entry and editing a little because Access must update the index whenever you add or change data. So you should index only the

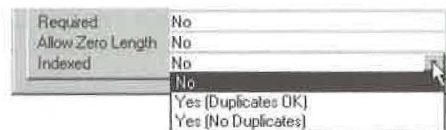
NOTE

If the field's data type allows it, the field isn't a primary key, and the field name starts or ends with *ID*, *key*, *code*, or *num*, Access will create a Yes (No Duplicates) index automatically. To specify which field names will trigger an automatic index, choose Tools ► Options, click on the Tables/Queries tab, and then edit or replace text in the AutoIndex On Import/Create box. See Chapter 15 for more details.

field(s) that you're most likely to use for sorting and searching. (To save you time and trouble, the Database Wizard and Table Wizard set up indexes on appropriate table fields automatically.)

To add or remove an index on a field:

1. Click on the name of the field you want to work with.
2. Click on the General tab under Field Properties.
3. Click on the Indexed property if it's available, click on the drop-down arrow next to Indexed, and then click on one of these options:



When defining a field as an index, be sure to choose Yes (Duplicates OK)—rather than Yes (No Duplicates)—unless you're absolutely sure that no two records in the table should have identical values in that field. To remove an index from a field, set the Indexed property to No. You'll see examples of indexed fields later in this chapter.

WARNING

The primary key automatically gets an Indexed property of Yes (No Duplicates). You can't change that setting unless you remove the primary key on that field.

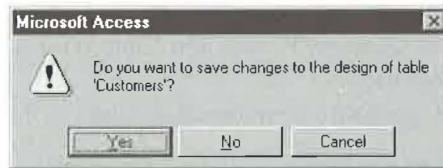
TIP

You can view and change the index name, field name, sort order, and index properties for all the table's fields at once if you wish. To do so, choose View ► Indexes from the menu bar or click on the Indexes toolbar button (refer to Figure 6.2).

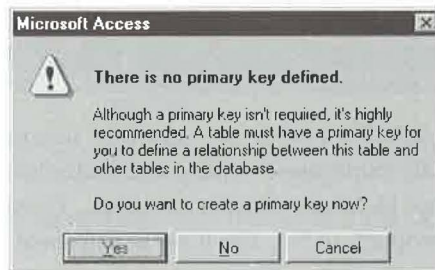
Saving a Table Structure

Once you're (reasonably) satisfied with the fields in your table, you can close and save the table structure. Here's how:

1. Choose File ► Close from the menu bar or press Ctrl+W or click on the Close button in the table design window. You'll probably see a dialog box similar to this one:



2. Click on Yes.
3. Type the name you want to assign to the table (up to 64 characters, including blank spaces) if you're prompted for a table name and then click on OK. If you haven't defined a primary key, you'll see this dialog box:



4. Define a primary key. If you're not sure how to answer, we suggest that you click on Yes and let Access create a primary key for you. (Access will create a field named ID with the AutoNumber data type.) Your database usually runs faster if every table has a primary key.

You'll be returned to the database window where you can see the new table name on the Tables tab.

TIP

You can save the table structure without closing the table design window first. Just click on the toolbar's Save button or press Ctrl+S or choose File ► Save anytime you're in the table design window. Access will save the changes you've made so far, and you'll remain in the table design window.

WARNING

When you save tables, forms, reports, queries, and other objects in your database, the database will increase in size. To make the current database smaller, you must compact the database. To get started, close the database, choose Tools ► Database Utilities ► Compact Database, specify the database to compact from, and then choose a database name to compact to. See Chapter 17 for more details.

Opening a Table

After you've created a table, you can open it anytime:

1. Click on the Tables tab in the database window.
2. Click on the name of the table you want to open.
 - To open the table for entering or editing *data*, click on the Open button or double-click on the table name.
 - To view or change the table's *design* (structure), click on the Design button.

You also can open or design a table by right-clicking on its name in the database window and then choosing Open or Design from the shortcut menu. The shortcut menu for tables also offers other handy options—Print, Print Preview, Cut, Copy, Save As/Export, Create Shortcut, Delete, Rename, and Properties. To find out the purpose of any option on the shortcut menu, point to that option with your mouse and look at the status bar for a brief description; then, if you need more details, press the F1 key.

Switching between Design and Datasheet Views

Once the table is open, you can switch quickly between *datasheet view* and *design view*. The differences between the two views are

- In **datasheet view** you typically work with the table's contents (data). However, you also can make some changes to the table's structure (more about this in Chapter 8).
- In **design view** you work with the table's structure only (field names, data types, properties), not with its contents.

To switch views while a table is open, click on the View toolbar button:



Switch from datasheet view to design view



Switch from design view to datasheet view

You also can switch to design or datasheet view by choosing View ► Design View or View ► Datasheet View from the menu bar. Or click on the drop-down arrow next to the View toolbar button and then choose Design View or Datasheet View.

Why Two Views?

You will use datasheet view to add data to your table as explained in Chapter 8. Figure 6.3 shows some names and addresses typed into a table named Customers. In datasheet view, field names appear across the top of the table, and none of the underlying structural information (such as data types and properties) is visible. If some fields are scrolled off the right edge of the window, you can use the horizontal scroll bar at the bottom of the window to scroll left and right through the fields.

FIGURE 6.3

The Customers table in datasheet view.

Customer ID	Company Name	Contact First N	Contact Last N	Billing Address
1	ABC Corporation	Wilbur	Stumingi	P.O. Box 555
2	RNAA Associates	Shirley	Ujest	187 Suffolk Ln.
3	Reese Clinic	Susita	Schumack	2817 Milton Dr.
4	WorldWide Widgets	Anita	Adams	6356 E. 20th St.
5	University of the Elite	Sheila	Stumingi	P.O. Box 23981
6	Precision Bagpipes	Andy	Adams	87 Polk St.
7	Database Search and	Hanley	Strappman	2743 Bering St.

AutoNumber)

Record: 8 of 8

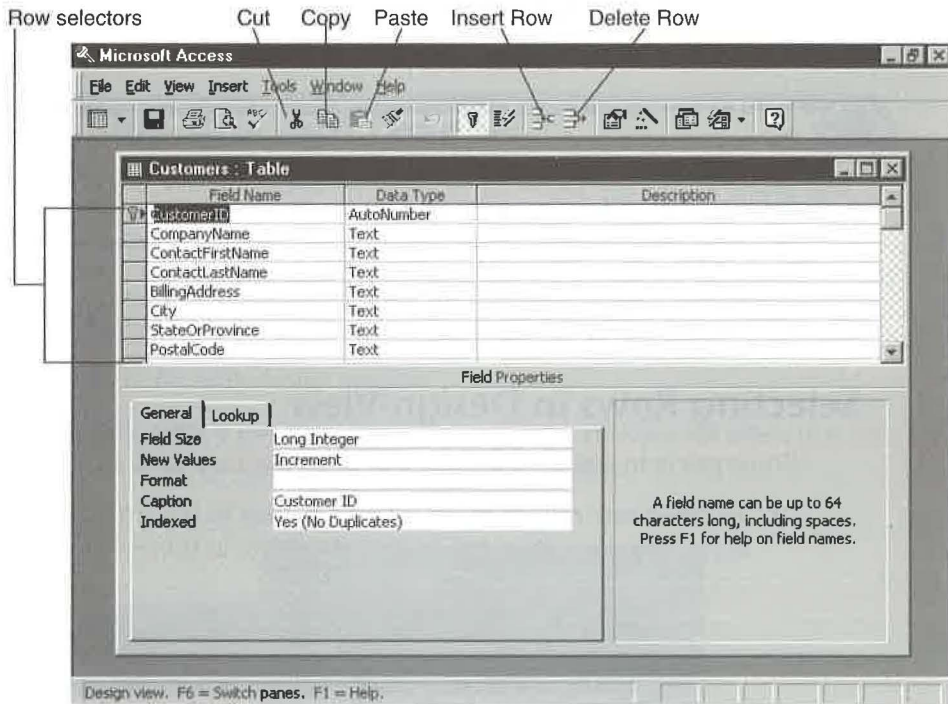


When adding data to a table that includes an AutoNumber field (such as the CustomerID field shown in Figure 6.3), Access usually will number records sequentially starting with 1 (1, 2, 3, and so forth). You can use the trick described under “Changing the Starting Value of an AutoNumber Field” near the end of this chapter to change the starting number for your table’s records (1001, 1002, and so forth).

In design view you can see and change the underlying table structure. Field names are listed down the left column, and the data type, description, and field properties appear across the table. None of the table’s data is visible in design view. Figure 6.4 shows the same Customers table in design view.

FIGURE 6.4

The Customers table in design view.



Modifying the Table Structure

Here are some techniques you can use to change a table's structure in design view. Open the table that you want to modify in design view (or switch to design view if you're in datasheet view).

- **To change a field's name, data type, description, or properties,** use the same techniques you used when creating the table structure.
- **To insert a new field into the structure,** move the cursor to where the field's new row should appear and then click on the Insert Rows toolbar button (shown at left) and press the Insert key; or choose Insert ► Rows from the menu bar.
- **To delete a field from the table structure,** move the cursor to the row that contains the field you want to delete and then click on the Delete Rows toolbar button. Or choose Edit ► Delete Rows from the menu bar.
- **To undo an accidental insertion or deletion,** click on the Undo toolbar button or choose Edit ► Undo or press Ctrl+Z.



WARNING

Do not rename or delete fields without carefully considering the effect of doing so. Queries, forms, reports, and other objects that rely on the presence of certain fields won't work properly if you rename or delete those fields.

You can also make some changes to a table's structure in datasheet view. See Chapter 8 for more information on how this works.

Selecting Rows in Design View

While you're in table design view, you can select rows (that is, fields) to manage:

- **To select one row,** click on the row selector to the left of the field you want to select. The row will be darkened (highlighted) as shown below:

	Field Name	Data Type	Description
1	CustomerID	AutoNumber	
2	CompanyName	Text	
3	ContactFirstName	Text	

- **To select multiple adjacent rows,** drag the mouse pointer through the row selectors of all the rows you want to select. Or click on the selector for the first row you want to select and then Shift-click on the selector for the last row you want to select.

- **To select multiple nonadjacent rows**, Ctrl-click on the row selector for each row you want to select.
- **To deselect selected rows**, click on any Field Name, Data Type, or Description box.
- **To deselect one selected row** in a group of selected rows, Ctrl-click on the row you want to deselect.

After selecting one or more rows, use these techniques to manage them:

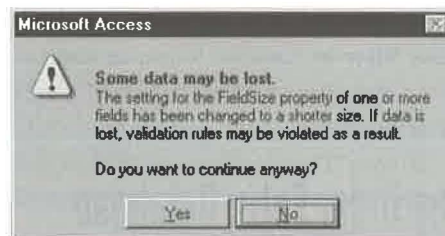
- **To move one selected row**, click on its row selector again, but this time hold down the mouse button and drag the selection to its new location.
- **To copy the selection**, choose Edit ► Copy or press Ctrl+C or click on the Copy toolbar button. Next click on the row selector for the empty row where you want to put the copy and then choose Edit ► Paste or press Ctrl+V (or click on the Paste toolbar button). Finally, *rename the copied field because no two fields in a table can have the same name.*
- **To delete the selection**, press the Delete key.
- **To insert blank row(s) above the selection**, press the Insert key.

Saving Table Structure Changes

After changing the table structure, you can save those changes in any of these ways:

- Switch to datasheet view or close the table design window (File ► Close). If asked whether you want to save the new structure, click on Yes if you want to save the changes or No if you don't want to save them.
- Choose File ► Save or press Ctrl+S or click on the Save toolbar button. You'll remain in the table design window.

If the table already contains data and your design changes will affect that data, a message will warn you of the change. Here's an example:



Read the warning carefully and then click on the appropriate command button.

Moving, Copying, Deleting, Renaming Entire Tables

You can easily copy, delete, and rename entire tables from the database window. The techniques are explained in Chapter 1, under “Managing Database Objects.”

Changing Properties of an Entire Table or Object

You can change the overall properties for a table or other Access object. For example, you can add a Description that will appear in the database window. Here’s how:

1. Go to the object’s design view window or click on the appropriate tab on the database window and then click on the object’s name. Different properties are available, depending on whether you start from the design window or the database window; however, you can change the Description property by starting from either window.
2. Choose View ► Properties from the menu bar or click on the Properties toolbar button (shown at left). Or if you’re starting from the database window, right-click on the object name and choose Properties.
3. Complete the Properties dialog box.



More Database Window Tricks

Viewing an object’s Description from the database window is a simple process: Choose View ► Details or click on the Details toolbar button or right-click on an empty area inside the database window and choose View ► Details.

You can sort objects in the database window by name, description, date created, date modified, or type. Any of these techniques will work:

- If you’re viewing the database window in detail (View ► Details), click on the column heading you want to use for sorting. For example, click on the Name column heading to sort objects into ascending (A–Z) alphabetical order by object name. Click on the column heading again to sort the objects in descending (Z–A) alphabetical order.
- Choose View ► Arrange Icons, or right-click on an empty area in the database window and choose Arrange Icons. Then choose By Name, By Type, By Created, or By Modified.

Tables in the Order Entry Database

Throughout this book, we’ll often refer to the Order Entry database, which we created using the Database Wizard described in Chapters 3 and 5. If you haven’t created a sample Order

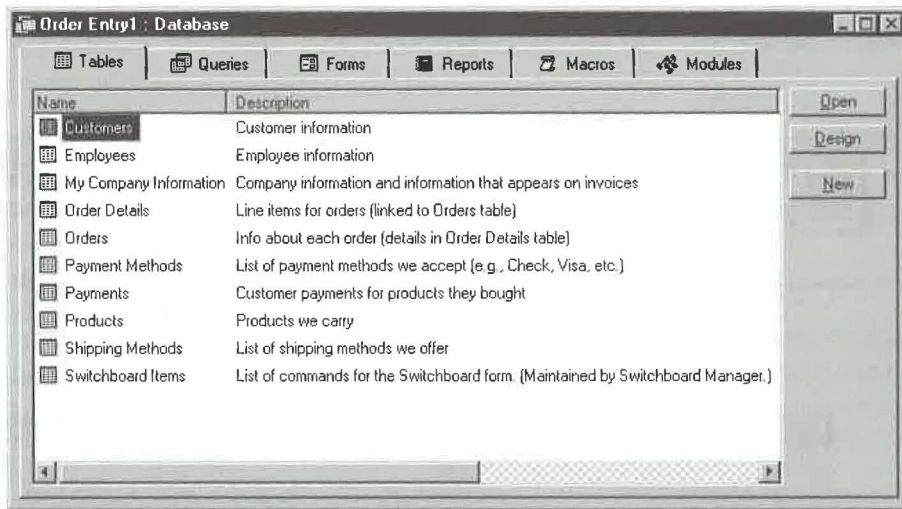


Entry database yet, you might want to do so now. Feel free to use the sample data the Wizard provides, your own data, or a combination of the two. Having this database available on your own computer will make it easier for you to follow the examples in this book.

Figure 6.5 shows the tables that make up the completed Order Entry database and describes them briefly. We'll give you more details about these tables in the sections that follow.

FIGURE 6.5

The database window showing all the tables in the Order Entry database. To explain the purpose of each table, we defined a Description property for each table.



The Customers Table

The Customers table is an important table in the Order Entry database because it stores name and address data for customers. Figure 6.6 shows this table's field names and data types.

Customers Primary Key and Index

The primary key in the Customers table is the CustomerID field. It's an AutoNumber field, which means that new entries are assigned an ID number automatically as they're added to the table. Furthermore, that number will never change (because the value in an AutoNumber field can't change), and no two entries ever have the same ID number. Like all primary key fields, CustomerID is indexed and duplicates aren't allowed.

FIGURE 6.6

The Customers table structure in the Order Entry database.

Field Name	Data Type
CustomerID	AutoNumber
CompanyName	Text
ContactFirstName	Text
ContactLastName	Text
BillingAddress	Text
City	Text
StateOrProvince	Text
PostalCode	Text
Country	Text
ContactTitle	Text
PhoneNumber	Text
FaxNumber	Text

NOTE

The New Values field property on the General tab offers you two ways to assign values to AutoNumber fields. The default choice, Increment, increases the field's value by 1 for each new record. To assign *random* long number values to new records, choose the Random option instead. Only one AutoNumber field is allowed in each table.

Changing Your Screen Display Size

Most figures in this book are shown with the Windows Desktop Area set to 640 by 480 pixels. However, we chose a desktop area of 800 by 600 pixels for some examples (such as Figure 6.5) that don't fit on a 640 by 480 screen. To change the Desktop Area on your own screen (if your

monitor supports such changes and you're running Windows 95), minimize or close the Access program window (if it's open), right-click on an empty area on the Windows desktop (not the Access window), choose Properties, click on the Settings tab, drag the Desktop Area slider to the appropriate setting, click on OK, and then answer any prompts that appear.

About Validation and Input Masks

Many people assign input masks, the Required property, and other restrictions to certain fields. For example, if you explore the General properties of the Customers table,

you'll notice that input masks are defined for the PostalCode, PhoneNumber, and FaxNumber fields as described below:

PostalCode The input mask 0000\-9999 allows you to enter zip+4 codes without typing a hyphen. *Example:* You type 857114747, and the input mask changes your entry to 85711-4747.

PhoneNumber The input mask !(999)"000\-0000 lets you enter phone numbers without having to type the parentheses around area codes or the hyphen after the exchange. *Example:* You type 6035551234, and the input mask changes your entry to (603)555-1234.

FaxNumber The input mask !(999)"000\-0000 is the same one used for the phone number.

NOTE

If you need to omit the area code when entering phone or fax numbers that have the !(999)"000\-0000 input mask, use the mouse or your keyboard to skip the area code portion of your entry.

You might want to remove these input masks if you want your own database to handle international names and addresses. Requiring a particular pattern, or requiring an entry at all, can prevent the entry of addresses in foreign lands. In fact, we don't even know the postal code or telephone number format used in Zimbabwe or Mongolia. Do you?

TIP

Restrictive field properties can make data entry frustrating or downright impossible. When you're first designing a table, avoid making the field properties too restrictive. You can always make the table more restrictive after you've worked with it for a while.

The Products Table

The Products table stores information about products sold by the business that uses the database. Its structure appears in Figure 6.7. The ProductID field is the primary key in the Products table to ensure that each product the business sells has a unique identifying code.

FIGURE 6.7

The Products table structure in the Order Entry database.

Field Name	Data Type
ProductID	AutoNumber
ProductName	Text
UnitPrice	Currency

The Orders Table

The Orders table tracks all orders placed. Ordering information is divided into two tables (named Orders and Order Details) because a natural one-to-many relationship occurs between a given order and the number of details (or line items) that go with that order (see Figure 6.8).

FIGURE 6.8

A natural one-to-many relationship occurs between an order and the number of items ordered (sometimes called *order details* or *line items*).

The screenshot shows two Access table views. The top view is 'Orders : Table' with columns: Order ID, Customer ID, Employee ID, Order Date, and PO Num. The bottom view is 'Order Details : Table' with columns: Order Detail ID, Order ID, Product ID, Quantity, and Unit Price.

Order ID	Customer ID	Employee ID	Order Date	PO Num
1	Let's Stop N Shop	Buchanan, Steve	2/2/97	52

Order Detail ID	Order ID	Product ID	Quantity	Unit Price
1	1	Crystal ball	2	\$45.55
2	1	Tether ball	5	\$9.65
3	1	Basketball	2	\$4.95
4	1	Football	2	\$5.65
5	1	Foosball	4	\$17.85

NOTE

The Order Entry database comes with an Orders By Customer form, which lets you enter and display data from several tables at once. You'll learn how to create similar data entry forms in Chapters 11 and 13.

Figure 6.9 shows the structure of the Orders table. The OrderID field is the primary key and is defined as the AutoNumber data type so that orders are numbered automatically as they're entered into the table. The fact that OrderID is an AutoNumber field will have implications for the Order Details table (as you'll learn shortly).

FIGURE 6.9

Structure of the Orders table in the Order Entry database.

The screenshot shows the 'Orders : Table' structure with columns and their data types.

Field Name	Data Type
OrderID	AutoNumber
CustomerID	Number
EmployeeID	Number
OrderDate	Date/Time
PurchaseOrderNumber	Text
ShipName	Text
ShipAddress	Text
ShipCity	Text
ShipStateOrProvince	Text
ShipPostalCode	Text
ShipCountry	Text
ShipPhoneNumber	Text
ShipDate	Date/Time
ShippingMethodID	Number
FreightCharge	Currency
SalesTaxRate	Number

The CustomerID field in the Orders table plays two important roles. First, it's an example of a field that uses Lookup properties. In this case the CustomerID field in the Orders table looks to the CustomerID field in the Customers table for a list of possible values. Lookup fields can make data entry quick and error free. (There's more about lookup fields later in this chapter.)

Second, the CustomerID field in the Orders table relates the Order to the CustomerID field in the Customers table, thereby ensuring that our database includes a valid customer record for each order placed. In the Customers table, CustomerID is an AutoNumber field. In the Orders table, the CustomerID field must have the Long Integer property for Access to match records correctly. A similar relationship exists between the OrderID field in the Orders table and the OrderID field in the Order Details table. (See "Defining a Relationship When One Field Is an AutoNumber" near the end of this chapter.)

Other interesting fields in this table include

EmployeeID A lookup field that lets you choose which employee sold the order.

ShippingMethodID A lookup field that lets you choose a valid shipping method for the order.

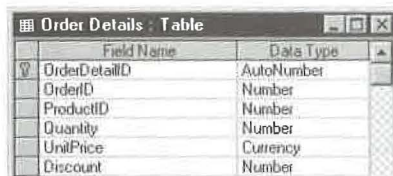
SalesTaxRate A number that's displayed in Percent format. A validation rule for this field requires entries to be less than 1 (<1) and displays an error message (validation text) if you enter a value greater than or equal to 1. *Example:* If you type .065 into the SalesTaxRate field, Access displays a value of 6.5%. If you type 6.5 into this field, Access instead displays the validation text error message "This value must be less than 100%."

The Order Details Table

The Order Details table stores one record for each line item of an order. Its structure is shown in Figure 6.10. In this table the OrderDetailID field is defined as the AutoNumber field type and as the table's primary key.

The OrderID field in the Order Details table relates each Order Details record to the appropriate record in the Orders table, which ensures that each line item belongs to a

FIGURE 6.10
Structure of the
Order Details
table in the
Order Entry
database.



Field Name	Data Type
OrderDetailID	AutoNumber
OrderID	Number
ProductID	Number
Quantity	Number
UnitPrice	Currency
Discount	Number

valid order. OrderID is the foreign key to the AutoNumber primary key field named OrderID in the Orders table. This field is indexed, with duplicates allowed, so that each order can have many line items.

NOTE

Foreign key is a fancy term for the table field(s) that refers to the primary key field(s) in another table. When the primary key has an AutoNumber data type, the foreign key field *must* be assigned the Number data type with the Long Integer field size. (See “Defining a Relationship When One Field Is an AutoNumber” near the end of this chapter for more details about foreign keys.) For other data types, the foreign key and its related primary key must have *exactly* the same data type and size.

Other interesting fields in the Order Details table include these:

ProductID A lookup field that lets you select a valid product from the Products table.

Discount Like the SalesTaxRate field in the Orders table, this field is displayed in Percent format and must be less than 1.

The Employees Table

The Employees table stores employee information. Figure 6.11 shows this table’s structure. Notice that the EmployeeID field is both an AutoNumber field and the table’s primary key. The WorkPhone field is defined with the usual input mask for phone numbers.

FIGURE 6.11

The structure for the Employees table.

Field Name	Data Type
EmployeeID	AutoNumber
FirstName	Text
LastName	Text
Title	Text
Extension	Text
WorkPhone	Text

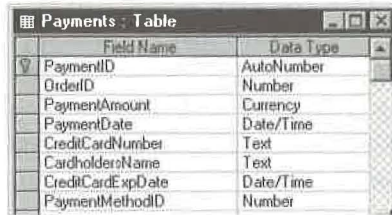
The Payments Table

The Order Entry application uses the Payments table to track which orders have been paid, how much of the order was paid, when the payment was made, and so forth. Figure 6.12 shows the structure for this table: The PaymentID is an AutoNumber field and the primary key for the table. Other interesting fields include the following:

OrderID Ties the payment to an order in the Orders table, and it is indexed (with duplicates OK).

FIGURE 6.12

The structure for the Payments table, which keeps track of customer payments.



Field Name	Data Type
PaymentID	AutoNumber
OrderID	Number
PaymentAmount	Currency
PaymentDate	Date/Time
CreditCardNumber	Text
CardholdersName	Text
CreditCardExpDate	Date/Time
PaymentMethodID	Number

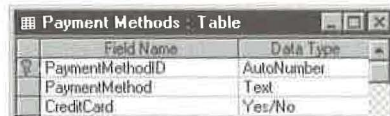
PaymentDate A Date/Time field that has an input mask of 99/99/00, which lets you enter dates such as 2/2/95 in any of these ways: 020295, 022/95, 2/2/95.

CreditCardExpDate A Date/Time field that has the same input mask as PaymentDate.

PaymentMethodID A lookup field that lets you choose a valid payment method from the Payment Methods table.

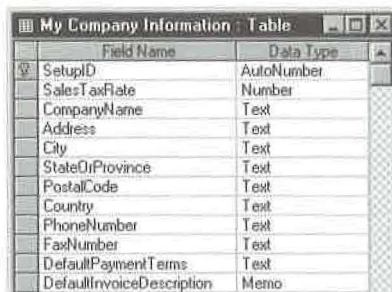
Other Tables in Order Entry

Order Entry uses two other small tables—Payment Methods (shown below) and Shipping Methods—which provide lookup information for the main tables already described.



Field Name	Data Type
PaymentMethodID	AutoNumber
PaymentMethod	Text
CreditCard	Yes/No

Finally, the table named My Company Information is used to define standard text and default settings for invoices. Its structure is shown below:



Field Name	Data Type
SetupID	AutoNumber
SalesTaxRate	Number
CompanyName	Text
Address	Text
City	Text
StateOrProvince	Text
PostalCode	Text
Country	Text
PhoneNumber	Text
FaxNumber	Text
DefaultPaymentTerms	Text
DefaultInvoiceDescription	Memo

In the My Company Information table, several fields have input masks that we've already described. The SalesTaxRate field has a validation rule and text that will make Access reject entries greater than or equal to 1. The PostalCode field has a 00000\-.9999 input mask, and the Phone-Number and FaxNumber fields have the !(999)"000\ -0000 input mask.

We've mentioned lookup fields many times in this chapter. Now we'll explain how to set them up in Access tables.

About Lookup Fields

You can create *lookup fields* that speed up and simplify data entry in a table's datasheet or in a form. As the name implies, the lookup field *looks up* values from another place and then automatically fills in the value you select. The source data can come from any of these places:

- A table that has a primary key field
- A query (or SQL statement) that displays specified columns and data from a table
- A fixed list of values that you enter when you create the field
- A list of all the field names in a table

For example, when entering data into the Order Details table, you can look up and fill in a ProductID field by selecting a value from a drop-down list. Figure 6.13 shows the ProductID field in the Orders form after we clicked on its drop-down arrow. To fill in (or change) the ProductID, we just click on an entry in the list.

FIGURE 6.13

The ProductID lookup field (shown here on the sample Orders form) lets you look up and fill in the ProductID with a click of your mouse. The data for this lookup field comes from the Products table.

Product	Unit Price	Quantity	Disc	Line To
Crystal ball	\$45.55	2	0.00%	\$91.10
Basketball	\$4.95	5	0.00%	\$24.75
Billiard balls	\$127.45	2	0.00%	\$254.90
Crystal ball	\$45.55	2	0.00%	\$91.10
Football	\$17.85			
Footbal	\$5.65			
Soccer ball	\$12.95			
Tether ball	\$9.65			
Volleyball	\$3.25			

Order Subtotal	\$231.95
Freight Charge	\$2.00
Sales Tax	\$0.00
Order Total	\$233.95
Total Payments	\$233.95
Amount Due	\$0.00

Record: 1 of 5

NOTE

In datasheet view, a lookup field always appears as a combo box, even if you've set its Display Control property to List Box in table design view. But when you add a lookup field to a form (such as the Orders form shown in Figure 6.13), Access automatically uses the Display Control property to decide whether to create the control as a combo box (which has a drop-down arrow that you must click to see the list) or as a list box (which shows the list itself, without a drop-down arrow).

How do you decide when to use a query, a table, a list of values, or a list of field names to display source data for your lookup field? The following sections offer some answers to this question.

Use a Query When...

You should use queries or SQL statements to display the list of values in a lookup field under the following conditions:

- You want columns in the lookup field's drop-down list to appear in a specific order (such as PersonID, LastName, FirstName).
- You want to display the results of calculations or to combine data from several fields into a single column. For example, a query or SQL statement makes it easy to display a customer or employee name as "Ajar, Doris" instead of showing the name in separate LastName and FirstName columns. (The EmployeeID field in the Orders table and Orders form displays employee names in LastName, FirstName format, as Figure 6.13 shows.)
- You want to restrict the lookup field's values to selected rows in a table. Suppose your Products table includes discontinued products that aren't available for sale. A query that filters out discontinued products offers a perfect way to display product names in the lookup field while preventing anyone from entering an order for discontinued products.
- You want changes to data in the query's source table(s) to be reflected in the drop-down list *and* in the lookup field that uses the query.

In a nutshell, queries and SQL statements are the best tools for making the lookup field display exactly what you want it to. They're also more efficient than using tables alone. You'll learn how to create queries in Chapter 10.

NOTE

An *SQL statement* is an expression that defines an SQL (Structured Query Language) command. Access uses SQL statements behind the scenes to interpret any queries you create in the query design window. SQL statements typically are used in queries and aggregate functions and as the record source for forms, reports, list boxes, and combo boxes that Access Wizards create. For more about SQL, look up *SQL Statements* and its subtopics in the Access Help Index.

Use a Table When...

You can use tables to display the list of values in a lookup field under the following conditions:

- Columns in the table are arranged in the order you want them to appear in the lookup field's drop-down list during data entry.
- You don't know how to create a query or SQL statement or you don't want to bother.
- You want changes to data in the source table to be reflected in the drop-down list and in the lookup field that uses the table.

Use a List of Values When...

Occasionally you might want to use a list of values when the list won't change often and it doesn't need to be stored in a table. Such a list might be handy for honorifics or salutations. Keep in mind that changes to data in the value list will be reflected in the drop-down list but not in any records you added to the lookup field before you changed the list value.

Use a List of All Field Names in a Table When...

This method is useful mainly to application developers. Suppose you're designing a data entry form for a user who knows little about Access, and that user wants the option to sort the table's records by any field he or she chooses. You could add a combo box or list box lookup field that displays all the fields in the table, place a control for that new field on your form, and attach to the control's *On Update* property a macro or event procedure that sorts the data by whatever field the user selects from the combo box or list box.

NOTE

If you want the user to sort by specific fields only, use a list of values that contains the sortable field names rather than a list of all field names in the table. The list of values method is preferable if the table you're planning to sort contains nonsortable field types, such as Memo or OLE Object fields.

If all this sounds like gobbledegook right now, don't worry about it. Access gives you plenty of ways to use lookup fields—even if you don't know about macros, event procedures, On Update properties, and the like. If you're curious about using these techniques, see Chapters 20 and 25.

Setting Up a Lookup Field

You can use two methods to set up a lookup field in Access. First, you can use the Lookup Wizard, which guides you through the process step by step. Second, you can use the Lookup tab in the Field Properties area of the table design window to define the lookup field manually.

With the Lookup Wizard...

When you're first learning how to set up lookup fields, you might want to stick with the Lookup Wizard. Here's how to use it:

1. Switch to the design view for your table.
2. Move the cursor to the empty row that should contain your new lookup field. You can leave the field name blank because the Lookup Wizard will assign a field name for you.
3. Click on the Data Type cell next to the blank field name, click on the drop-down arrow, and then choose Lookup Wizard. The Lookup Wizard will take over so follow the prompts in the dialog boxes.
4. Choose whether to get values from an existing table or query or whether to type the values you want. Click on the Next button.

TIP

As usual, you can click on the Next button to move to the next step, the Back button to back up to the previous step, and the Cancel button to bail out early.

5. Complete the remaining dialog boxes, reading each one carefully and clicking on the Next button to move ahead. Be sure to choose or define column names in the order you want to see them in the drop-down list during data entry. Keep these questions and answers in mind as you work through the Lookup Wizard dialog boxes:
 - **Which columns do you see in the drop-down list during data entry?** By default, the columns will appear in the order you chose them in the Lookup Wizard (though you may need to use the scroll bars to see them all); however, the primary key column usually will be hidden. You can change this default behavior in the Lookup Wizard dialog box that lets you adjust column widths. To do so, uncheck the Hide Key Column box or hide any column by dragging (to the left) the vertical line next to the column name until the column disappears or reposition any column by clicking on its column name and then dragging the name to the left or right.
 - **What relationships are created behind the scenes?** If data for your new field comes from a table, the Lookup Wizard creates a relationship between the current table and the table whose data you're looking up. The field you're creating becomes the foreign key in that relationship. This behind-the-scenes action will ensure that you don't accidentally delete the lookup table or lookup fields without first deleting the relationship.
 - **What values appear in the table after you choose an item from the drop-down list during data entry?** The *first* column you leave visible will appear in the table.
 - **What values are actually stored in the new field when data comes from a related table?** If you've chosen to hide the related table's key column, the new field will store values from the *primary key field* of the related table. If you've chosen not to hide that table's key column, the field name you select in the second-to-last Lookup Wizard dialog box controls which data is stored in the new field. (This dialog box asks you to choose a field that uniquely identifies the row.)
 - **What values are actually stored in the new field when data comes from a query or a list of values you type in?** The field name you select in the second-to-last Lookup Wizard dialog box controls which data is stored in the new field.
 - **What name is assigned to the new field?** If data for your new field comes from a table or query, the Lookup Wizard gives the new field the same name as the primary key of the related table or the first field of the query. It sets the new field's Caption property to whatever name you choose in step 6 below. If data for your new field comes from a list of values you typed in, the Lookup Wizard uses the name you assign in step 6 for the field name and leaves the Caption property blank.

6. Type a label to use as the caption for your new field (if you wish) in the last dialog box and then click on the Finish button to finish the job. Respond to any prompts that appear.

In addition to handling all the details described in step 5 above, the Lookup Wizard also fills in the Lookup properties for you automatically. To see these properties for yourself, go to the table design window, click in the row that contains your lookup field, and then click on the Lookup tab in the table design window's Field Properties area.

With the Lookup Properties Sheet...

Of course, you can define Lookup field properties without a Lookup Wizard, as follows:

1. Switch to the design view for the table.
2. Specify the Field Name and Data Type for your lookup field if you haven't done so already.
3. Click on the Lookup tab in the table design window's Field Properties area.
4. Click in the Display Control property box, click on the drop-down arrow that appears, and then choose either List Box or Combo Box.
5. Complete the remaining properties in the Lookup property sheet (see the next section for details).

Figure 6.14 shows the Lookup properties for the EmployeeID field of the Orders table. Figure 6.15 shows the query that's "behind" the SQL statement shown in the Row Source box of Figure 6.14. To display this query, we clicked in the Row Source box on

FIGURE 6.14

The Lookup properties for the EmployeeID field of the Orders table.

Store this column's value in the EmployeeID field.

Click on the ... button to view or change the SQL query that supplies data for the field's combo box.

Display this many columns in the field's combo box.

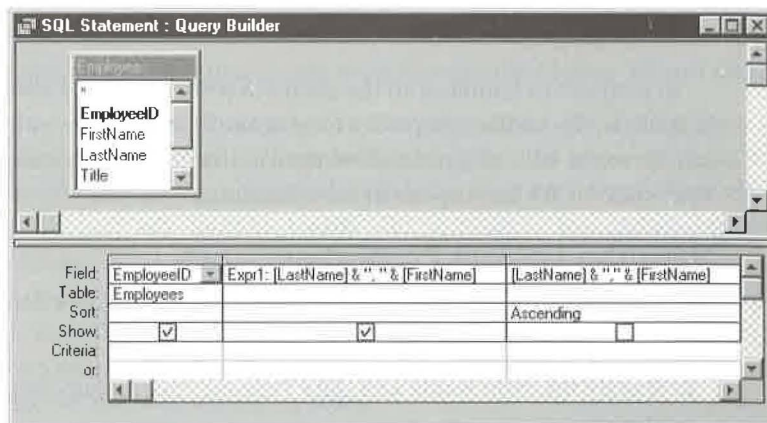
Display each column with the specified width.

Field Name	Data Type	Description
CustomerID	Number	
EmployeeID	Number	
OrderDate	Date/Time	

Property	Value
Display Control	Combo Box
Row Source Type	Table/Query
Row Source	Select [EmployeeID].[LastName] & "" & [F...]
Bound Column	1
Column Count	2
Column Heads	No
Column Widths	0";2"
List Rows	8
List Width	2"
Limit To List	Yes

FIGURE 6.15

The query behind the SQL statement for the Row Source property shown in Figure 6.14.



the Lookup properties tab, clicked on the Build button that appeared, and widened the second column in the query design grid. Figure 6.16 shows the results of running that query. (We ran this query so you can see the relationship between the query results and the Lookup properties.) See Chapters 3 and 10 for more about queries.

FIGURE 6.16

The results of running the query shown in Figure 6.15.



Important Facts about Lookup Fields

You'll have great success with setting up lookup fields if you remember these points:

- If you plan to use a table or query to supply values for the lookup field, that table or query must exist. (When using the Lookup tab, rather than the Lookup Wizard, you can create a query on the fly by clicking in the Row Source property box and then clicking on the Build button. See Chapter 10 for more about queries.)

- If you're using a table to supply values for the lookup field, that table must have a primary key.
- You can hide a column in your drop-down list by setting its width in the Column Widths property box to 0.
- The first column in the drop-down list is always the one you see in the lookup field (unless you've hidden the column by setting its column width to 0; in this case, the column you see is the first one that isn't hidden).
- The value that's actually stored in the lookup field is the one that's specified as the bound column (more about this shortly). The data type of the lookup field must be compatible with the bound column, regardless of what you see in the field during data entry.

NOTE

The last two points above can be confusing. The easiest way to keep them straight is to remember that the first (unhidden) column controls *what you see* when you scroll through the table in datasheet view. The bound column controls *what actually is stored* (what you get) in the lookup field.

With these points in mind, let's take a closer look at the properties on the Lookup tab in the table design window.

Understanding the Lookup Properties

At first glance, the property sheet for the Lookup tab might seem rather daunting (see Figure 6.14). But it's not bad once you know more about it. Here's the scoop:

Display Control Lets you choose the type of control used to display the field on the datasheet and forms. (For lookup fields, set the Display Control property to List Box or Combo Box.)

Row Source Type Lets you choose the type of source for data in the field (Table/Query, Value List, or Field List). Generally, you'll want to choose Table/Query.

Row Source Lets you choose the source of the control's data. In Figure 6.14 the row source is the SQL Select statement, which expands to the query shown in Figure 6.15.

Bound Column Lets you specify which column in the row source contains the value to store in this lookup field. The data type for the field must be compatible with values in the bound column. For instance, if the bound column stores numeric values (such as EmployeeID values), the data type for the lookup field also must be numeric (Number, Currency); if the bound column stores long integer values, the data type for the lookup field must be either Long Integer or AutoNumber.

Column Count Lets you specify how many columns to display. The example in Figure 6.14 tells Access to display two columns; however, the first column is hidden because its Column Widths setting is 0.

Column Heads Lets you choose whether to show the column headings. By default, no column headings appear. To show column headings, change the setting to Yes.

Column Widths Lets you specify the width of each column shown in the drop-down list. In Figure 6.14, the first column width is 0 (hidden), and the second column is 2 inches wide. When typing the column widths, use a semicolon (;) to separate each column's width specification; you can omit the unit of measurement (""). For example, if you type 0;2 in the Column Widths property box, Access will change your entry to 0";2" when you move the cursor out of the Column Widths box.

List Rows Lets you specify how many rows to display in the drop-down list at once (the default is 8).

List Width Lets you specify the width of the entire combo box. (A List Width of *Auto* tells Access to calculate the width automatically.)

Limit To List If set to Yes, only values shown in the drop-down list are allowed during data entry. If set to No, Access allows entries that aren't shown in the drop-down list.

For immediate help as you define Lookup properties, click in the appropriate property field and press F1.

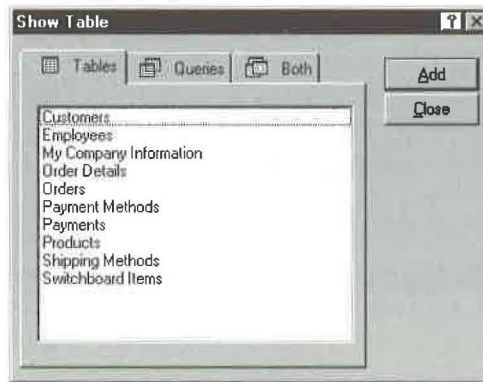
Defining Relationships among Tables

You can define all the relationships among your tables at any time. There are several advantages to defining the relationships early on (before you add much data):

- When you open multiple related tables in a query (see Chapter 10), the related tables will be joined automatically, saving you a few extra steps.
- Access will create some needed indexes automatically to make your related tables perform more quickly.
- You can define *referential integrity* relationships between tables when joining them. Referential integrity ensures that the relationships between records in related tables remain valid and can prevent problems from occurring when you try to delete or change a record that's related to records in another table. *Example:* Referential integrity can ensure that every Orders record has a corresponding record in the Customers table (thus every order can be traced back to a customer). It also can prevent you from deleting a customer record if outstanding orders remain for that customer.

To define relationships among existing tables:

1. Close any open tables so that only the database window is visible.
2. Choose Tools ► Relationships from the menu bar, click on the Relationships toolbar button (shown at left), or right-click anywhere on the database window and choose Relationships. The Relationships window appears. (It will be empty unless you've previously defined relationships among your tables.)
3. Display tables for which you want to define relationships by clicking on the Show Table toolbar button (shown at left); by right-clicking on an empty part of the Relationships window and choosing Show Table; or by choosing Relationships ► Show table from the menu bar. You'll see a dialog box similar to this one:



4. Add a table or query to the Relationships window by clicking on an appropriate tab (Tables, Queries, or Both) in the Show Table dialog box. Then click on the name of the table or query you want to add and click on Add; or double-click on the table or query name. To add several tables at once, use the standard Shift-click and Ctrl-click techniques to select the tables and then click on the Add button.
5. Repeat step 4 until you've added all the tables and queries for which you want to define relationships. Then click on the Close button.
6. Relate the tables as explained in the next section.

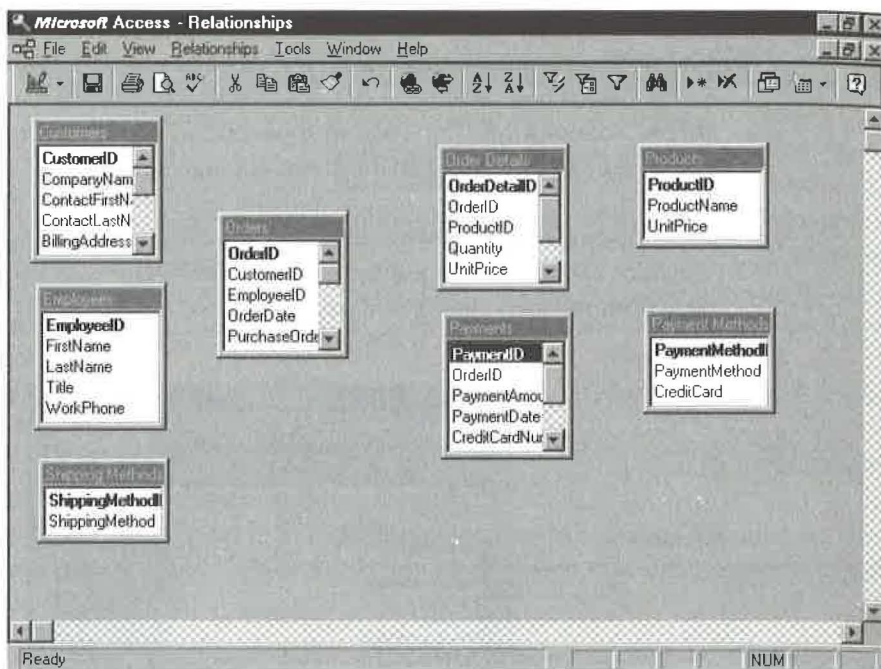
Figure 6.17 shows tables from the Order Entry database that are good candidates for relating. We've arranged and sized the tables, but haven't shown the relationships among them.

TIP

To tidy up the Relationships window anytime, move the tables by dragging their title bars, and resize the tables by dragging their borders.

FIGURE 6.17

Tables from the Order Entry database added to the Relationships window. The relationships among the tables haven't been defined yet.

**NOTE**

If you created the Order Entry database with the Database Wizard, all the relationships among the tables will be defined for you automatically, and the window will look more like Figure 6.18 than Figure 6.17 (although you'll certainly need to rearrange the tables to straighten out the spaghetti appearance of the lines between them).

Relating Two Tables

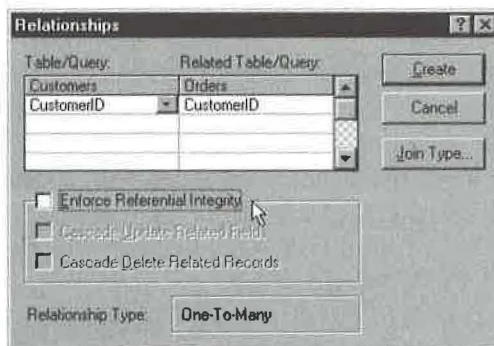
To define the relationship between any two tables in the Relationships window:

1. Move the mouse pointer to the primary key field in the *primary table* (the table on the "one" side of a one-to-many relationship). That key is boldfaced in the list.
2. Drag that field name to the corresponding field in the related table (that is, drag it to the appropriate *foreign key*).



You can drag from the foreign key field in the related table to the primary key field in the primary table. The results will be the same.

3. Release the mouse button to display a dialog box similar to this one:



4. (Optional) Select (check) the Enforce Referential Integrity box to enforce referential integrity between the two tables. Then you may tell Access how to handle changes and deletions in the primary table:
 - If you want *changes* to the table on one side of the relation to automatically carry over to the related table, select (check) the Cascade Update Related Fields checkbox.
 - If you want *deletions* in one table to carry over to the related table, select the Cascade Delete Related Records checkbox.



In order to define referential integrity between tables, the matching field from the primary table must be a primary key or have a unique index, the related fields must have the same data type (or be AutoNumber and Long Integer), and both tables must be stored in the same Access database. For more details about referential integrity rules, go to the Relationships dialog box and press the F1 key.

5. (Optional) Change the type of join between tables by clicking on the Join Type button, choosing the type of join you want to use, and then clicking on OK. (If in doubt, don't make any change. You can always define the relationship in a query, as discussed in Chapter 10.)

NOTE

Join is a database term that describes the correspondence between a field in one table and a field in another table. You can perform several types of joins, though the default join (called an *equi-join* or *inner join*) works fine for most situations. To learn more about joins, search for *Joins, Defining Types For Tables* in the Access Help Index.

6. Click on the Create button to finish the job.

Access shows the relationship between the two tables as a join line connecting the related fields. The appearance of the line indicates the type of join you've chosen and whether you're enforcing referential integrity. In the example below, the thick, solid bars in the join line indicate that referential integrity is enforced between tables. The small 1 indicates the table on the one side of this relationship; the small infinity sign indicates the table on the many side of this relationship.



You can repeat steps 1 through 6 above to define the relationships between as many pairs of tables as appropriate in your database. Figure 6.18 shows all the relationships between tables in the Order Entry database as the Database Wizard defined them.

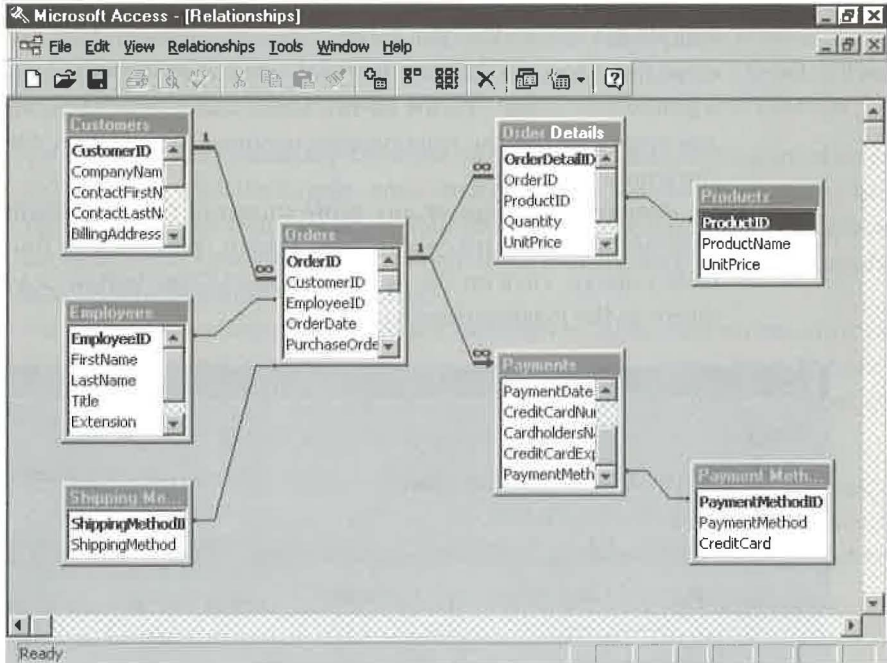
In the Order Entry database, referential integrity with cascaded updates and deletions is enforced between these tables: Customers and Orders, Orders and Payments, Orders and Order Details, and Products and Order Details. To make data entry more flexible, referential integrity is not enforced between these tables: Shipping Methods and Orders, Payment Methods and Payments, and Employees and Orders.

WARNING

When you're first designing a database, don't go hog wild with setting up relationships between tables. If the relationships between tables are too strict, data entry can be cumbersome. It's easy to add or remove relationships later if you need to, so don't worry too much about defining every possible relationship. Start conservatively, add a few sample records, and then decide whether the relationships should be more restrictive or less restrictive.

FIGURE 6.18

The Order Entry Relationships window, showing all relationships the Database Wizard defined.



More Tips for Using the Relationships Window

Here are some other tips for using the Relationships window:

- To show tables that are directly related to a selected table, click on the table and then click on the Show Direct Relationships toolbar button (shown at left). Or right-click on the table you're curious about and choose Show Direct. Or click on the table and choose Relationships ► Show Direct.
- To show all tables and relationships that are currently defined for the database, click on the Show All Relationships toolbar button (shown at left). Or right-click on an empty area in the relationships window and choose Show All. Or choose Relationships ► Show All from the menu bar.
- To hide a table that you've added to the Relationships window, click on the table you want to hide. Then press Delete; right-click and choose Hide Table; or choose Relationships ► Hide Table from the menu bar. This step does not delete the table from the database, and it doesn't delete any relationships defined for the table; it just hides the table from view. (To add the table to the view again, click on the Show Table toolbar button, double-click on the appropriate table name, and click on Close.)



- **To clear all tables from the Relationships window**, choose Edit ► Clear Layout or click on the Clear Layout toolbar button (shown at left). When asked for confirmation, click on Yes. The Relationships window will be empty. Again, no tables or relationships are deleted, but your work toward tidying up the Relationships window *will* be lost. To see all the tables and relationships again, right-click on any empty part in the Relationships window, choose Show All, and then tidy up the window as needed.
- **To change the design of any table** shown in the Relationships window, right-click on that table and choose Table Design. When you're finished changing the table's design, click on the design window's Close button or press Ctrl+W; you'll return to the Relationships window.

Printing the Relationships Window and other "Unprintable" Things

Alas, there's no way to print the Relationships window directly. Your best bet is to set up the window the way you want it

and then press Alt+Print Screen to capture the screen to the Windows Clipboard. Then start the Paint applet (Start ► Programs ► Accessories ► Paint) and choose Edit ► Paste or press Ctrl+V. Finally, print the image from Paint by choosing File ► Print ► OK.

Saving the Relationships Layout

When you use the Relationships window to add, change, or delete relationships, the relationships are saved automatically; however, changes to the window's layout are not. Therefore, any time you're happy with the appearance of the Relationships window, you should save the layout. There are many ways to save a layout: Choose File ► Save or press Ctrl+S or click on the Save toolbar button or right-click on an empty part of the Relationships window and choose Save Layout. If you want to save the layout and close the Relationships window in one step, choose File ► Close or press Ctrl+W or click on the window's Close button; then click on Yes when asked about saving your changes.

After you've defined all the relationships among your tables, close the Relationships window (choose File ► Close). If you're asked about saving changes, click on Yes.

Redefining (or Deleting) a Relationship

If you later discover that you've made a mistake while defining the relationships among your tables, you can follow the same steps presented under "Relating Two Tables" to return to the Relationships window and view existing relationships.

- **To change the relationship between two tables**, double-click on the thin part of the join line and then make your changes in the Relationships dialog box. Or right-click on the thin part of the join line and choose Edit Relationship from the shortcut menu. Or click on the thin part of the join line and choose Relationships ► Edit Relationship from the menu bar.
- **To delete the relationship between two tables**, right-click on the thin part of the join line that you want to delete. Then choose Delete from the shortcut menu. Or click on the thin part of the join line and then press Delete or choose Edit ► Delete. When prompted, click on Yes to confirm the deletion.

For more examples and information on defining relationships between tables, open the Microsoft Access Contents, open *Creating, Importing, And Linking Tables*, open *Defining Relationships And Setting Referential Integrity Options*, and then explore the subtopics. You also can search the Help Index for *Relationships* topics and subtopics.

Important Tips for AutoNumber Fields

AutoNumber fields are great time savers in Access. But you must understand how they work in order to use them effectively. To help you get the most from using AutoNumber fields, we'll first explain how to set a starting value for AutoNumber fields. Then we'll explain how to define a relationship between tables when one field's data type is AutoNumber.

NOTE

AutoNumber fields aren't always the best choice for assigning meaningful identifiers, such as customer numbers, because you can't delete or change AutoNumber field values. Furthermore, the autonumbering sequence will contain "holes" where you've deleted records. For example, if you've entered three customer records, the AutoNumber fields will have the values 1, 2, 3. If you then delete the customers whose AutoNumber fields are 2 and 3, the AutoNumber field for the next new customer will have the value 4. Sometimes this will be desirable, and sometimes it won't.

**WARNING**

If the table for which you're changing AutoNumber field values contains data and that table is involved in a referential integrity relationship, you may lose records from related tables (if cascaded deletes are allowed)—or you may not be allowed to do step 13 below (if cascaded deletes are not allowed). Therefore, you should do the steps in the next section *only* if your related tables are empty (or contain only a few records that you don't care about).

Changing the Starting Value of an AutoNumber Field

Suppose you want to use an AutoNumber field to uniquely identify each customer in a table, bumping each new customer number by 1. However, you want the numbering to start at some number other than 1, such as 1001 for four-digit ID numbers or 10001 for five-digit numbers. Here's how to change the starting value when you are in design view:

1. Open the table that contains the AutoNumber field. (Make sure the AutoNumber field's New Values property on the General tab is set to Increment.)
2. Choose File ► Save As, click on *Within the Current Database as*, enter a new name (such as *_Temp*), and then click on OK. (Make sure the title bar of the table design window reflects this new name.)
3. Change the data type of the AutoNumber field from AutoNumber to Number (in design view for the new table); make sure its Field Size property is set to Long Integer.
4. Switch to datasheet view and click on Yes when prompted for permission to save the table.
5. Type the starting number you want to use for the AutoNumber field, *minus 1*. (You also must enter a value in any required fields you've defined.) In the example below, we assigned the first customer an ID of 1000 so that our customer numbers for "real" data will begin at 1001.

Customer ID	Company Name	Contact First Name	Contact Last Name	Billin
1000				
0				

6. Close the temporary table (press Ctrl+W).
7. Highlight the name of that temporary table in the database window and choose Edit ► Copy.
8. Choose Edit ► Paste.

9. Type the name of the *original* table in the Table Name box (Customers in our example).
10. Click on the Append Data To Existing Table option button.
11. Click on OK to complete the copy.
12. Double-click on the name of the original table in the database window (Customers in our example) to view its contents. You should see both the original record(s) (if any) and the newly numbered one(s), as shown below.

Customer ID	Company Name	ContactFirstName	Contact Last Name
1	ABC Corporation	Andy	Adams
2	RNAA Associates	Hanley	Strappman
3	Reese Clinic	Susita	Schumack
4	WorldWide Widgets	Shirley	Ujest
5	University of the Elite	Wilbur	Stumingi
6	Precision Bagpipes	Sheila	Stumingi
7	Database Search and Rescue	Anita	Adams
1000			
(AutoNumber)			

13. Skip to step 15 if the table doesn't contain old records. Otherwise, move the record(s) with the old numbers to the Windows Clipboard. To do this, drag the mouse pointer through all the record selectors *except* the selector for the new record you just added, choose Edit ► Cut (or press Ctrl+X), and then click on Yes.
14. Paste the records back into the table by choosing Edit ► Paste Append ► Yes. The new records will start with the next available AutoNumber value (1001 in this example).
15. Click on the record selector for the empty temporary record (1000 in this example), press Delete, and then click on Yes to delete the record.
16. Add at least one new record to the table if the table was empty after you finished step 15 and you plan to compact the database that contains the table (see Chapter 17). Adding this new record will prevent the compacting operation from resetting the value for the next record added to one more than the previous value (effectively undoing your efforts in the above 15 steps).
17. Close the table and click on No if asked about keeping the data on the Clipboard.
18. Highlight the name of the temporary table (_Temp), press Delete, and then click on Yes as needed to delete it.

NOTE

This procedure will work *only* if the temporary table's starting number (e.g., 1000) is at least one higher than the highest AutoNumber field value that's currently stored in the table you're renumbering.

Any new record that you add to the table will be numbered starting at one more than the last record currently in that table.

Defining a Relationship When One Field Is an AutoNumber

When you define a relationship between two tables and the primary key on the one side of the relationship is an increment AutoNumber field, the foreign key (that is, the corresponding field on the many side) must be the Number data type with its field size set to Long Integer. Figure 6.19 shows an example.

NOTE You also can match a Replication ID AutoNumber field with a Replication ID Number field. For information about replication, see Chapter 17 or look up *Replication* in the Access Answer Wizard.

FIGURE 6.19

When the primary key on the one side of a relationship is an AutoNumber field, the corresponding field on the many side must be a Long Integer.

Primary key: The field that uniquely identifies each record on the one side of a one-to-many relationship.

When the primary key is the AutoNumber data type...

...the foreign key must be the Number data type with its Field Size property set to Long Integer.

Field Name	Data Type	script
CustomerID	AutoNumber	
CompanyName	Text	
ContactFirstName	Text	
ContactLastName	Text	
BillingAddress	Text	
City	Text	
StateOrProvince	Text	
PostalCode	Text	

Field Name	Data Type	script
OrderID	AutoNumber	
CustomerID	Number	
EmployeeID	Number	
OrderDate	Date/Time	
PurchaseOrderNumber	Text	
ShipName	Text	
ShipAddress	Text	
ShipCity	Text	

Property	Value
Field Size	Long Integer
New Values	Increment
Format	
Decimal Places	
Input Mask	
Caption	Customer ID
Indexed	Yes (No Duplicates)

Property	Value
Field Size	Long Integer
Format	
Decimal Places	Auto
Input Mask	
Caption	Customer ID
Default Value	
Validation Rule	
Validation Text	
Required	No
Indexed	Yes (Duplicates OK)

Primary table: The table on the one side of a one-to-many relationship.

Foreign key: The field in the table on the many side that relates each record to a record in the primary table.

The reason for this requirement is somewhat technical, but basically it's because Access stores AutoNumber field data as four-byte numbers. Only a foreign key that's exactly the same four-byte number will match the primary key. Setting the field size property of a number to Long Integer ensures that a four-byte number will be stored in the field.

If you forget to set the property of the foreign key to Long Integer, you won't be able to define a relationship between the tables. Or worse yet, if you add data to both tables without defining a relationship first and then try to access related records from both tables, the results will be incorrect and very confusing.

This requirement is just one of those picky little things you need to remember about Access. If you forget and problems arise much later down the road, you might not think to check the field properties in the tables until you've spent 20 hours on hold with technical support and pulled out all your hair.

Documenting Your Tables and Database

Having a printed copy of information about your table structures can be very handy when you start developing other objects in your database. Chapter 17 covers these "database administration" tools in detail. But here's a quick preview of techniques for the curious:

- **To print design information about any database objects** (including tables or the entire database), choose Tools ► Analyze ► Documenter, or click on the drop-down arrow next to the Analyze button on the database window's toolbar and choose Documenter.
- **To view or change database properties**, choose File ► Database Properties, or right-click on the database window's title bar or any gray area on the database window and choose Database Properties.
- **To print the table design window or the Relationships window**, use standard copy/cut-and-paste techniques to copy the screen to the Windows Clipboard, paste the Clipboard contents into Windows Paint, and then print the results. See the sidebar titled "Printing the Relationships Window and other Unprintable Things" earlier in this chapter.

Analyzing Your Tables and Database Performance

Two Access Wizards can help you optimize your tables and improve database performance. You'll learn about both of these in Chapter 16. But for now, here's a summary of what's available:

Table Analyzer Wizard Analyzes any table you choose and lets you split it into related tables, if necessary. To get started, choose Tools ► Analyze ► Table, or click on the drop-down arrow next to the Analyze button on the database window's toolbar and choose Analyze Table.

Performance Analyzer Wizard Analyzes relationships, tables, queries, forms, reports, macros, modules, or all objects, and provides suggestions for making the selected objects work more efficiently. To use this Wizard, choose Tools ► Analyze ► Performance, or click on the drop-down arrow next to the Analyze button on the database window's toolbar and choose Analyze Performance.

Where to Go from Here

In this chapter, you've learned how to design tables using both the Table Wizard and various from-scratch methods. Where should you go from here?

- To learn how to import or link existing data into your tables, see Chapter 7.
- To learn more about entering data into your tables, see Chapter 8.

What's New in the Access Zoo?

The process of creating and managing tables with Access 97 for Windows 95 is the same as it was in earlier versions of Access. The big news is the new Hyperlink

field type, which lets you store hyperlink addresses in tables. Hyperlinks are used to jump to all kinds of information: different database objects, other Microsoft Office documents, or even data on the Internet or an intranet.





A black and white photograph of a mountain range. The mountains are covered in snow and have a rugged, layered appearance. In the foreground, there are dark, dense evergreen trees. In the upper right corner, a full moon is visible in the dark sky. The overall scene is serene and majestic.

Chapter

7

Linking, Importing, and
Exporting Data



FEATURING

Learning about linking and importing 210

Importing and linking tables 213

Importing and linking spreadsheets and text files 225

Importing and linking HTML files 237

Troubleshooting import problems 239

Exporting data from Access 242

Using the Publish to the Web Wizard 247
