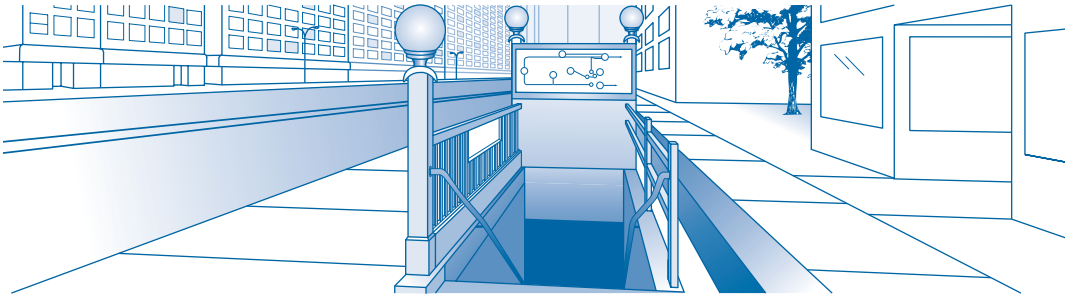
CHAPTER

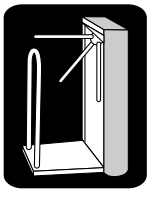
1

Designing Databases

Stations Along the Way

- Exploring the anatomy of a database
- Designing a database
- Using a naming convention
- Creating a database
- Deleting a database





Enter the Station

Questions

1. What is a database?
2. What are the parts of a database?
3. What is a relational database?
4. How do you create a database in Microsoft Access?
5. What's the most basic element of database design?
6. How do you delete a database?
7. How can you tell whether a database is well-designed?

Express Line

If you already understand the basics of database structure and design, then skip ahead to Chapter 2.

This chapter guides you through an introduction to databases, the parts of a database, some design principles, a little database theory, and practical examples. In this chapter, you can figure out how to design, name, create, and delete databases.

Discovering Databases

A *database* is an object that stores related data. In some database programs, the database is simply a folder on your hard drive, but in Microsoft Access, it's much more. An Access database is a specialized container that holds such information as data, tables, macros, modules, forms, and reports. Most database systems don't provide such a feature-rich environment. In most other database programs, a database is simply a storage-and-retrieval engine for large amounts of table data, and other programs handle all the remaining features.

Microsoft Access provides a database *client* (the program that shows you your data) and a database server, plus data-entry forms that you create, reports for summarizing and displaying data, and two programming languages.

Information Kiosk

Technically, Microsoft Access isn't a database server. However, multiple clients can connect to it simultaneously, like a server. You can share your database with anyone who has the Microsoft Access program. Several people can work on the same database at the same time.

Information Kiosk

A database program is usually referred to as a Database Management System (DBMS) or as a Relational Database Management System (RDBMS). Though most people use the terms *database* and *DBMS* interchangeably, I use *database* to mean a single container of data; *DBMS* refers to Microsoft Access itself. I don't use *RDBMS* at all because Access is a relational database, so the relational part of the definition is assumed.

You can think of a database (in general terms) as a collection of similar or related data. Typically, when you create a database, you have some idea of the type of data that you plan to store in it — all that data should have something in common. For example, you could put all the data for your company in one database and data about all the books in a library in another database, but you might not want to put all the data of both types into the same database. Trying to keep different types of data in one database may cause your single database to grow to an unmanageable size.

In the following section, I show you the basic database-design principles that let your databases grow and progress normally — and point out how to avoid some common database pitfalls.

Designing a Database

Before you open Access and start putting things in database files, take some time to consider what you want to put in your database — and the ways you want to use that information. The more carefully you've thought out a design for your database *before* you build it, the more useful Access will be to you.

Design principles

The most basic design element of a database is the consistent naming of its components. Simply put, a *naming convention* is a way of categorizing and managing the items in a database by creating and using descriptive labels for those items.

This naming process may sound very simple, but you must decide on a naming convention for your database work and stick to that convention throughout your project. Consistency and clarity are very important attributes of any naming convention.

Information Kiosk

A naming convention is a standard way of naming and referring to items such as databases, tables, forms, reports, and so on. You must decide for yourself on the naming convention for your projects. Although there's no standard naming convention that works for all databases, it's a good idea to keep the names short and descriptive — for example, you might use `Publishers` for a list of book publishers and `Titles` for book titles in a `Books` database.

Although nobody has created ironclad standards for naming database objects, the next section offers a few guidelines for coming up with your own consistent naming convention — and sticking to it. Note that the idea here is to create *scalable* names (able to grow with the project); they'll be easier to migrate to a higher-end database product such as Microsoft SQL Server if you have to scale up to a client/server database at some point.

Practical rules for scalable naming conventions

A naming convention should follow these rules:

- Easy to decipher
- Consistently capitalized

- Descriptive of the data
- Relatively short

A naming convention should avoid these pitfalls:

- Don't use *spaces.*
- Don't be *cryptic* or use a *foreign language.*
- Don't use *numbers* unless they're significant in some logical way (such as the date, part of the name of a company, or a musical group).
- Don't use any characters except *capital letters* (A–Z), *lowercase letters* (a–z), *numbers* (0–9), *underscored spaces* (_), and *hyphens* (-).

For example, if you name a database `Guestbook`, your naming convention has the first letter capitalized and the remainder of the name in lowercase. This type of naming convention is common — but what if you have two or three different guest books, each representing a different company? Well, you could name them `Guestbook1`, `Guestbook2`, and `Guestbook3` — but those numbers don't offer any information about which company is using which guest book. Instead of numbering the guest books, you could put company initials in your naming convention — something like `Guestbook_DBD`, `Guestbook_SMS`, and `Guestbook_KDS`. These descriptive names give you immediate clues about the guest-book data contained in each database.

Remember, if you have fifty guest books and two companies have the abbreviated name of `DBD`, then you should change your naming convention accordingly. For example, say you're creating databases for both Delta Boat Disassemblers and Data Bank Dynamics. You could expand their guest-book names to `Guestbook_Delta_Boat_Dis` and `Guestbook_Data_Bank_Dyn`.

You want to group similar data into a single unit and keep it separate from other (unrelated) data — and a database is the primary object for grouping data. In the preceding example, you wouldn't put salary information in a `Guestbook` database designated to hold guest data such as names, addresses, e-mail addresses, and comments.

Step into the Real World

You may find the best way to organize your ideas about a database project is to write down your thoughts about organizing data on paper before starting with Access. The following section describes the process you can go through in detail.

Step into the Real World

A poorly designed database may work just well enough to get the job done — many sub-par databases are being used every day — but why accept it? This chapter (and this book) shows how to build well-designed databases that you can use for the life of your project.

Accounting data offers a good example. Suppose you created a database named `Accounting` that contained tables with such names as `Accounts_Rec`, `Accounts_Pay`, `Payroll`, and `Taxes`. Here you're using separate tables for accounts receivable, accounts payable, and so on — but not separate databases. It's a matter of scale: the Accounting department is usually kept separate from other departments within a company, so it gets its own database. The Human Resources department is another good example of a data grouping. A database called `Human_Resources` may have such tables as `Employees`, `Management`, `Executives`, `Salaries`, `Pension`, and `401K` — but all of these belong in the same database because they're all Human Resources concerns.

Information Kiosk

Create databases sparingly. Many companies follow the practice of creating a single database that contains many tables. You can also create databases that break up the data into logical subcategories, such as Accounting, Human Resources, and Manufacturing (for an automaker) or Stamps, Books, and Grades (for a school).

Organize your data

Creating a database can be as simple as saying, "I need a database for all of my recipes," and so you create a database and name it `Recipes`. You can create a database for your postage stamp collection and name that database `Stamps`, but the way to group your data is often far less obvious. That's when the process becomes a little more complicated.

Say you're creating a database for your company, Delta Boat Disassemblers, to use to keep track of employee information. You create a database named `DBD`, and inside the database, you decide to create a list of tables that will store employee data of all kinds, including salaries, insurance, Human Resources information, customer data,

sales, and inventory information. This scenario may work just fine, but you should consider many issues before deciding to take this approach, the greatest of which is organization of the data.

Putting the concept to work

The design of your database is absolutely critical to your success.

The following list shows my design progression, which you can use to create a database. These steps are effective and efficient:

1. **Write the name and purpose of the database.**

Watch Your Step

Always keep the purpose of your database in mind. By remembering why you're creating your database, you're less likely to include faulty or unrelated data. For instance, if you create a database to store all your recipes, you probably don't want to add your stamp or music collection to that database. The data within a database should be related in some logical way.

2. **List the major categories of data.**

   These categories usually become the tables within the database.

3. **List the elements within the categories.**

   These elements can become the columns in each table.

4. **Remove redundancies.**

Watch Your Step

Removing redundancies is the most important step. Chapter 3 covers removing redundancies in detail.

5. **Add and remove categories and elements.**

   Add categories for items that appear often in your data (if you haven't already) and clean up by removing any unneeded categories or elements.

6. **Create the database, tables, and columns.**

Creating a Database

Access requires that you either open an existing database or create a new one upon opening the program.

To create a new database when first opening Access, follow these steps:

1. **Start Microsoft Access.**

   When Access is open, your screen should look like Figure 1-1.

2. **Click Blank Database.**

   The right side of the application updates with a default database name of `Database1.accdb`. You can accept this name or use another name.

3. **Click Create to create the database and open it in Microsoft Access.**

Watch Your Step

Don't change the extension of the database (`.accdb`). If you change the extension, Access automatically appends the name with `.accdb`.

Figure 1-1: Getting started with Microsoft Access.

You can also create a new database within another Access database. Follow these steps:

1. **Click the Microsoft Office icon.**
2. **Click New.**

   The Getting Started with Microsoft Office Access page opens, as shown in Figure 1-1. The right side of the application updates with a default database name of `Database1.accdb`. You can accept this name or use another name.
3. **Click Create to create the database and open it in Microsoft Access.**

Deleting a Database

You can delete a database in two ways:

- As you would any other file on your computer
- From within Access

To delete a database from within Access, follow these steps:

1. **Click the Microsoft Office icon (in the upper-left corner of the program).**
2. **Click Open.**

   Your list of available databases appears in the window.
3. **Right-click the database file you want to delete.**
4. **Select Delete from the menu.**
5. **Confirm the deletion by clicking Yes.**

   To delete a database from outside of Access, simply drag the icon to the Recycle Bin or highlight the file name and press Delete.

Watch Your Step

You can't delete a database that's open. When you delete a database (inside or outside of Access), you can recover it only if it's still in your Recycle Bin.

The true purpose of a database is to store and retrieve data quickly. It's a centralized repository of information that's organized in such a way that you can easily access that information. Data doesn't generally get changed within a database, so you need to make sure that the data is of very high quality when you enter it into the database. Data is of *high quality* if it doesn't have redundancies, mistakes, or *anomalies* (faulty data).

database: An object that stores related data.

DBMS: Database Management System refers to a program for managing and storing data. Microsoft Access is a DBMS.

naming convention: A method of using consistent and easily deciphered names for database objects.

object: A database or any part of a database (such as a table).

RDBMS: Relational Database Management System is a DBMS that has the added feature of allowing objects, such as tables, to share data with each other. Microsoft Access is an RDBMS.

relational database: A database whose tables share common data elements with each other, making those tables related.

Practice Exam

1. Which of the following is the best definition of a database?

A) A collection of objects with similar names

B) A storage facility for data

C) An object that contains data

D) An object that stores a collection of related data

2. You have three notebooks filled with information that you want to put into Access. One notebook contains the names and addresses of your friends and family, another contains information related to your home-based business, and the third contains information about your music collection. When using Access to organize and store this data, you would:

A) Create a database named Info and put the data from all three notebooks in it.

B) Create a database and put only your business information in it because databases are far too important to store personal data.

C) Create two databases, one named Personal and one named Business, and put your addresses and music in Personal and your business info in Business.

D) Create three databases, one for each of the three general types of information.

3. Which of the following would be the best name for a database that holds information for the Destructo Demolition company?

A) `Destructo`

B) `DDemo`

C) `DeDeCo`

D) `Destructo Demolition Company`

4. Which of the following steps is the most important when designing a new database?

A) List the elements within the categories.

B) Remove redundancies.

C) Add and remove categories and elements.

D) Create the database, tables, and columns.

5. When developing a naming convention for your databases, you should:

A) Use simple, easily decipherable names.

B) Be creative.

C) Use cryptic names.

D) Use very long and very descriptive names.

6. True or false: You can create a new database from within Access. If true, explain how.

_____

_____

_____

7. If you delete a database accidentally, you may be able to retrieve it if:

A) No changes have been made to the database since it was deleted.

B) The database is less than 50 megabytes in size.

C) It's still in the Recycle Bin.

D) It has the appropriate security locks.

8. What's the true purpose of a database?

A) To organize data

B) To stick with a naming convention

C) To store and retrieve data

D) To locate faulty data

9. **True or false: A database should contain data that's related. Explain why or why not.**

_____

_____

_____

10. **Explain why you shouldn't name three new databases Stuff, Miscellaneous, and New.**

_____

_____

_____