用 Access 2000 和 Visual FoxPro 6.0 管理公司和个人数据

肖勇波　梁　冰　编著

清 华 大 学 出 版 社

需要全本请访问 www.ertongbook.com

(京)新登字 158 号

内 容 简 介

本书介绍了两个比较流行的数据库管理软件：Microsoft Access 2000 和 Visual FoxPro 6.0。两种软件在数据管理方面功能都非常强大，但是管理的风格大不相同。在本书中，作者分别以两个大的实例（设备管理和人事管理）详细地介绍利用两种工具管理数据的知识。读者既可以从中学习有关数据库管理的知识，又可以把理论应用到实践，增强读者的实际应用能力。本书的特点是不需要读者有太多的数据管理方面的知识，就能轻易地管理自己身边的各种数据。

本书面向的读者对象主要是：Access 2000 和 Visual FoxPro 6.0 的初学者和提高者，尤其适合于需要管理个人数据和公司数据的数据管理人员。

版权所有，翻印必究。
本书封面贴有清华大学出版社激光防伪标签，无标签者不得销售。

书　　名：用 Access 2000 和 Visual FoxPro 6.0 管理公司和个人数据
作　　者：肖勇波　梁　冰　编著
出 版 者：清华大学出版社(北京清华大学学研大厦，邮编 100084)
　　http://www.tup.tsinghua.edu.cn
责任编辑：欧振旭
印 刷 者：
发 行 者：新华书店总店北京发行所
开　　本：787×1092 1/16　印张: 20.75　字数：452　千字
版　　次：2001 年 5 月第 1 版　2001 年 5 月第 1 次印刷
书　　号：ISBN 7-302-04431-7 / TP • 2606
印　　数：0001～5000
定　　价：29.00 元

读者意见反馈卡

请您认真填写本卡并寄给我们。对于发现本书中技术问题的读者，我们另有答谢。

1．您对本书的总体感觉：

□满意　　□一般　　□不满意

2．您认为本书的层次结构：

□很好　　□一般　　□不好

3．您认为本书的语言文字水平：

□很好　　□一般　　□不好

4．您认为本书的版式编排：

□很好　　□一般　　□不好

5．您认为本书中所涉及各项操作说明的准确性：

□准确　　□较准确　　□不准确

6．您最需要哪方面的图书？ _____

7．您是从哪里第一次听说这本书的？

□书店　　□广告　　□从朋友、同事等处听说　　□其他

8．您一年中购买计算机类图书的数量：

□2～5 本　　□6～10 本　　□多于 10 本

9．您使用的操作系统是：

□DOS　　□Windows　　□OS/2　　□Macintosh　　□Unix　　□Linux　　□其他

10．您感兴趣的计算机类新书为：

□操作系统类　　□办公软件类　　□程序设计语言类

□图形、图像设计类　　□排版软件类　　□网络技术类

□多媒体制作类　　□其他

11．您使用 PC 机的地方：

□家庭　　□单位　　□学校　　□其他

12．您是否有 CD-ROM：

□有　　□无

读者姓名：　　单位名称：　　联系电话：

请填好本卡后寄给：

清华大学校内金地公司(E-mail：jdsd@public.east.net.cn)

《用 Access 2000 和 Visual FoxPro 6.0 管理公司和个人数据》编辑部收

邮编：100084

联系电话：（010）62770381　　传真：（010）62770384

如需本书可与本编辑部联系邮购（汇款请按以上地址填写，另加邮费 10%）

前　言

当计算机已经悄悄地走进人们的工作和生活中的时候，您是否想到，利用计算机来管理公司和个人的各种数据呢？目前市场上已经出现了各种各样的数据库管理软件，您可以充分地利用计算机的优势，将传统的手工管理的数据交给计算机来看管。您将发现，利用计算机来管理数据，更加方便、简洁，能够大大提高数据管理的准确度和管理的效率。

本书面向的对象主要是处于上班族的读者。对于上班族中的您来说，计算机已经是一个不可缺少的办公工具了，那么您可以充分利用现有的资源，来对您工作、生活中的数据进行有效的管理。本书向您介绍了两个数据库管理软件：Microsoft Access 2000 和 Visual FoxPro 6.0。

由 Microsoft 公司开发的 Access 数据库起初作为 Office 系列的应用软件，以其与 Office 其他应用软件相似的界面和友好的可视化操作而得到 Office 用户的喜好，Access 随着 Office 在企业应用中的普及而越来越多地为企业管理人员所接受，越来越多的人认为有必要学习有关 Access 应用和操作的知识。

FoxPro 作为数据库管理软件早就为中国用户所认识，从最初的 dBase、Foxbase 发展到 FoxPro，FoxPro 的风格已经为许多企业用户所接受，而由 Microsoft 公司推出的具有可视化界面的 Visual FoxPro 后，对于 FoxPro 数据库的操作就更加简单和方便了。但为了能够满足以前使用非可视化的 FoxPro 的用户的操作需要，Visual FoxPro 在给出菜单操作方式的同时，仍保留了原有的命令操作方式，这构成了 Visual FoxPro 有别于其他数据库操作软件的特点。

两种软件在数据管理方面功能都非常强大，但是管理的风格大不相同，您可以根据自己的爱好，选择一种自己所喜欢的管理工具。

同时，这两种软件也是当前中国企业中为用户所广泛使用的最为流行的数据库软件，将这两个数据库软件集中到一本书中介绍，可以使用户更加方便地了解到这两种软件之间的相似之处以及各自的特点和区别。同时掌握这两种软件的操作方法，也可以使得用户在日常应用中更加得心应手、有着更加广泛的适应面。

第一部分：Microsoft Access 2000

Access 2000 是 Microsoft Office 2000 系列软件之一。它是新一代关系型数据库管理软件，是多媒体与开放性数据库系统，具有直观性、开放及功能强大的特点。Access 2000 在 Access 97 的基础上进行了大量的改进，不仅界面美观，更重要的是功能更加强大。Access 2000 新增了网络的功能，使得它与 Internet 的结合更加方便。

Access 2000 是一个中、小型的数据库管理系统，非常适合管理公司和个人的各种数据。Access 2000 有一个最大的优势是具有非常好的用户友好性，它不需要用户有很

多的数据库基础，对于各种操作都提供了比较完善的操作向导，用户可以在向导的指导下，轻松、直观地进行各种操作，而其中涉及到的 SQL 语句的书写、Visual Basic for Application（VBA）代码的编写等工作，都由 Access 2000 自动来完成。这样大大减轻了用户的开发劳动，提高了数据管理的效率。

在本书中，我们用 12 章的篇幅比较系统的介绍了如何利用 Access 2000 来管理公司和个人的各种数据。首先，我们介绍了有关数据库方面的一些基础知识，这对于不熟悉数据库、数据管理的读者来说，可以增进对数据管理的直观认识；接下来，我们以一个实际的例子（建立设备管理系统），详细介绍了如何创建数据库，如何创建数据库中的各种对象等操作，在学习的过程中，根据作者的经验，我们给出了一些操作上的技巧和小窍门；最后，作为 Access 2000 学习的高级篇，我们介绍了 Access 数据库安全管理的问题，介绍了如何导入、链接和导出 Access 数据库中的数据，以实现和其他应用程序之间的数据交换，同时还介绍了如何对 Access 数据库进行加密等知识。

当然，Access 2000 的功能绝对不止本书所介绍的功能。在学习的过程中，读者会发现，Access 2000 已经涉及到了数据管理的方方面面。不过，对于处于上班族中的读者来说，利用书中所介绍的知识，来管理公司和个人的数据已经足够了。如果用户有兴趣，完全可以充分地挖掘，去探索 Access 2000 深层次的功能，从而使得所创建的数据库管理系统更加完善。

前言

第二部分：Visual FoxPro 6.0

Visual FoxPro 是当前较为流行的关系数据库可视化开发工具之一，它以卓越的数据库处理性能和良好的可视化开发环境赢得了广大用户的喜爱。Visual FoxPro 6.0 中文版是 Visual FoxPro 系列中最新推出的版本，它除了继承旧版的特点和优势外，还增加了许多新的功能，为数据库系统的开发提供了更快的速度、更强的能力和更大的灵活性。

本书面向初学者，共分为 12 章。第 13 章到第 22 章依次介绍了 Visual FoxPro 6.0 中文版的菜单命令、数据类型、自由表和数据表、查询及视图、报表、表单及表单控件的创建和使用方法，在介绍中引用同一数据库实例，便于读者领会和掌握。第 23 章和第 24 章简要介绍了 Visual FoxPro 的程序编制及面向对象的初步设计方法，供有进一步学习能力的读者阅读。

由于篇幅所限，本书主要着眼于让读者能迅速掌握最基本的概念和操作方法，内容简洁、图文并茂、通俗易懂。

本书中，Access 2000 的学习由肖勇波执笔完成，Visual FoxPro 6.0 部分的作者是梁冰，同时也得到了曹庆的大力援助。由于写作时间仓促，作者水平有限，书中不足之处，还请读者多多指正。

编　者

2000 年 5 月

需要全本请访问 www.ertongbook.com

第一部分　Microsoft Access 2000

第 1 章　数据库基础

本书的宗旨是向读者介绍如何利用数据库管理软件 Access 2000 和 FoxPro 来管理公司和个人的各种数据。处于上班族中的您可能已经非常熟悉数据的概念了，因为在您的工作中,每天都不可避免要和大大小小的数据打交道，或者您所从事的工作就是对公司的数据进行管理。那么在具体学习如何管理数据之前，读者有必要知道究竟什么是数据库，利用数据库可以管理公司和个人的什么数据。在这一章中，我们通过简要地介绍数据库的基础知识，来回答这些问题。如果读者对数据库的基本概念、关系数据库的基本原理已经有了一个比较全面的认识，可以跳过这一章，直接进入 Access 2000 和 FoxPro 的学习。

1.1　数据管理的方式

数据管理，实际上就是指对数据进行分类、组织、编码、存储、检索以及维护。这里数据包括的概念非常广泛，大到公司的各种相关记录（比如财务帐目、产品生产记录、公司人员档案等），小到个人的一些零碎的备忘录，比如约会备忘录、个人的通讯录等。我们每天都要和各种数据打交道，记录一些新的数据、查找所要的数据等等操作就属于数据管理的范畴。

数据有三种管理的方式：

1．手工管理方式

这是一种最早为人们使用，也使用得最为广泛的一种数据管理的方式，在手工管理方式下，记录数据的载体是纸张。在计算机还没有普遍利用之前，各个公司都采用定制的帐单表格来记录公司的各项财会数据，这种方式比较简单，不需要任何的硬件条件；但是它存在一些不可避免的缺点，比如没法保证数据记录不出现任何差错，效率低下，进行总帐统计时非常麻烦等等。手工数据管理只适合于数据量小，对数据要求不高的场合。事实上在计算机高度发达的今天，手工数据管理仍然广泛地存在着。比如在我们的日常生活中，利用备忘录、通讯录来手工记录一些小规模的数据仍然是相当方便的。

2．计算机文件管理方式

自从计算机深入人们的生活，很多人开始探索计算机的潜力，充分利用计算机的优势，利用计算机文件的形式来管理工作、生活中的某些数据。利用文件管理数据总是与一定的计算机软件相联系的。比如在 Windows 产品中提供了一个“记事本”，我们可以非常方便地利用它建立一个文本文件，记录一些备忘信息；我们也可以用一个电子表格（比如 Excel 表格）来记录我们的通讯录。记录的这些信息都以文件的形式存储在计算机中，需要查找信息的时候，只要打开文件即可。

利用计算机文件管理数据的一个最大的优点是查找方便、更新容易，并且在需要时可以打印出来，比起传统的备忘录、通讯录等要美观得多。但是用它来管理一些大规模的数据可能就缺乏效率了。

3．数据库管理方式

数据库管理实际上也属于文件管理，但是它是利用文件管理数据的高级形式。数据库可以简单地理解为数据的“仓库”，它包含有大量的数据。当需要记录的数据结构复杂，数据量日趋增多，对数据处理的要求越来越高时，利用数据库来管理数据就显得非常有效了。如今数据库技术已经发展得非常完善了，也出现了各种各样的数据库管理软件，利用数据库管理数据具有如下一些特点：

- 面向全组织的复杂的数据结构

数据的结构化是数据库的主要特征之一，也是数据库与文件系统的根本区别。在前面的文件管理方式中，文件是面向某一特定的目的的，而利用数据库，可以从公司全局的角度来设计所有数据的结构，这样可以对公司数据进行集中的管理，方便不同的使用部门。当然，这就要求在描述数据时不仅描述数据本身，还要求描述数据之间的关系。

- 数据冗余度小，容易扩充

由于数据库是从整体的观点来看待和描述数据的，数据不再是面向某一个特定的目的，而是面向整个系统，这可以大大减小数据的冗余度，既节约存储空间，减少计算机的存取时间，又可以避免数据之间的不相容性和不一致性。

- 具有较高的数据和程序的独立性

在管理数据库的过程中，可能需要对某些数据进行一定的统计、打印等操作，这可以通过编制一定的程序来实现。但是程序和数据库之间是相对独立的。如果对数据库的结构进行了调整，只要数据的逻辑结构不发生变化，所编制的应用程序也就没有必要发生变化了。正是因为数据库和应用程序之间的相对独立性，用户在编制应用程序时不必考虑数据库的具体存取路径等细节，从而简化了应用程序的编制，大大减少了应用程序的维护和修改。

- 数据的统一控制功能

数据库是整个公司中各用户的共享资源，为了防止不合法的使用所造成的数据泄露和破坏，在数据库中可以采用一定的安全保密措施（比如设置密码、对数据加密等）；为保证数据在输入、修改过程中的有效性，还可以在数据库中定义一定的有效性规则，在输入数据有误时，系统自动提出警告信息。

综上所述，可以说数据库是个通用化的综合性的数据集合。它可以供各种用户共享使用，并且具有最小的冗余度和较高的数据与程序的独立性。它适用于管理公司和个人的结构复杂、规模较大的各种数据，也是目前使用得最为广泛的一种数据管理方式。

1.2 数据库系统

数据库系统为我们提供了一种把与我们的工作和生活紧密相关的信息集合在一起的方法，它还提供了在某个集中的地方存储和维护这些信息的方法。数据库系统主要由三大部分组成：数据库管理系统（DBMS，它是专门负责组织和管理数据信息的程序）、数据库应用程序（它使我们能够获取、显示和更新由 DBMS 存储的数据）、数据库（按一定结构组织在一起的相关数据的集合）。

一般来说，DBMS 和数据库应用程序都驻留在同一台计算机上并在同一台计算机上运行，很多情况下两者甚至结合在同一个程序中，以前使用的大多数数据库系统都是用这种方法设计的。但是随着 DBMS 技术的发展，目前的数据库系统正向客户/服务器模式发展。客户/服务器数据库将 DBMS 和数据库应用程序分开，从而提高了数据库系统的处理能力。数据库应用程序运行在一个或多个用户工作站（客户机）上，并且通过网络与运行在其它计算机上（服务器）的一个或多个 DBMS 进行通信。

1.2.1 数据库管理系统

为了能够有效地及时地处理数据，并提供数据库的安全性和完整性，必须有一个软件系统——数据库管理系统（DataBase Management System，DBMS），以在建立、运用和维护数据库时进行统一的控制。在本书中即将向读者介绍的 Access 2000 和 FoxPro 就是目前市场上比较流行的两种数据库管理系统。

数据库管理系统是用于描述、管理和维护数据库的程序系统，是数据库系统的核心组成部分。它建立在操作系统（比如 Windows 2000、Windows NT 等）的基础上，对数据库进行统一的管理和控制。其主要功能有：

- 描述数据库：描述数据库的逻辑结构、存储结构、语义信息和保密要求等。
- 管理数据库：控制整个数据库系统的运行，控制用户的并发性访问，检验数据的安全、保密与完整性，执行数据检索、插入、删除、修改等操作。
- 维护数据库：控制数据库初始数据的装入，记录工作日志，监视数据库性能，修改更新数据库，重新组织数据库，恢复出现故障的数据库。
- 数据通信：组织数据的传输。

1.2.2 数据库应用程序

数据库管理系统中存储了大量的数据信息，其目的是为用户提供数据信息服务，而数据库应用程序是用来与 DBMS 进行通信，并访问 DBMS 中的数据的。它是 DBMS

实现其对外提供数据信息服务这一目的的惟一途径。简单地说，数据库应用程序是一个允许用户插入、修改、删除并报告数据库中的数据的计算机程序。数据库应用程序在传统上是由程序员用一种或多种通用或专用的程序设计语言编写的，但是近年来出现了多种面向用户的数据库应用程序开发工具,这些工具可以简化使用 DBMS 的过程，并且不需要专门编程。在 Access 2000 中，不能生成单独的可执行的应用程序，因为 Access 数据库将数据表、窗体、报表等对象统一包容在了数据库中；而 FoxPro 则可以编译、生成独立于数据库的可执行的数据库应用程序。

1.2.3 数据库

从定义上来讲，数据库就是信息的集合，指在计算机存储设备上合理存放的结构化的相关数据的集合。这种集合是与特定的主题和目标相联系的，例如，客户订单数据库、企业资产数据集合等。

数据库中数据之间是相互关联的，具有一定的对应关系。相互关联的数据之间，简单地说只有三种关系：即一对一、一对多和多对多。在这三种关联关系的基础上，构成了复杂多样的数据关系模型。在各种数据库管理系统软件中，主要有三种数据关系模型：层次模型、网状模型和关系数据库模型。前两者也称为非关系数据库模型，由于在存取数据上具有很多的不便，现在已经不常用了。目前关系数据库应用最为广泛，Access 2000 和 FoxPro 就是用来管理关系数据库的。接下来我们将重点对关系数据库中的几个概念进行介绍。

1.3 关系数据库

关系数据库是用以二维表格来组织和存放数据的，一个二维表就是一个关系，这种二维表可以存放两种信息，一是事物本身的信息，二是事物之间联系的信息。一般来说，一个关系数据库是由若干个表组成的，它包括表的结构及表中的数据两部分。

1.3.1 表

一个关系表（Table）就是一组相关的数据按行排列，像一张表格一样。比如一个班所有学生的期末考试成绩，存在一个表中，每一行对应一名学生，在这一行中，包括学生的学号、姓名以及各门课程的成绩，如表 1-1 所示。

表 1-1 学生成绩表

学号	姓名	语文成绩	数学成绩	外语成绩
950001	张三	60	86	90
950002	李四	70	78	86
950003	王五	80	65	78

在表中，每一列称为一个**字段**（Field）。每一个字段都有相应的描述信息，如数据类型、数据长度等。而每一行称为一条**记录**（Record），用于记录一个学生的完整信息。

一个关系表可以用“表名（字段 1，字段 2……）”的形式来表示。比如上面的学生表可以表示为：

学生（学号，姓名，语文成绩，数学成绩，英语成绩）

数据库中的关系有以下性质：

- 列是同质的，即每一列中的分量是同一类型的数据，来自同一个数据集合。
- 不同的列可出自同一个数据集合，每一列的字段名不能相同。
- 列的顺序无所谓，即列的次序可以任意交换。
- 任意两个记录不能完全一样。
- 行的顺序无所谓，即行的次序可以任意交换。
- 每一分量必须是不再可分的数据项。

在上面已经说到，在关系表中，不能存在两条完全一样的记录，我们一般选择表中的某个字段（又称为属性）或者几个字段的组合作为表的**主码**（又称主关键字），用主码的取值来惟一地标识一条记录。比如在表 1-1 中，一所学校可能出现名字相同的学生，但是每个学生的学号是惟一的，不可能相同，因此可以用学号来惟一地标识一个学生。

注意：

主码字段的取值不能为空。

在关系数据库中，还有一个很重要的概念——**索引**（Index）。索引是和数据在计算机存储器（比如磁盘）上的物理存取方式息息相关的，其作用和图书中的书目索引差不多，可以用来加快数据访问的速度。因此，在设计数据库时，可以根据需要，设定合理的索引字段。

与表相对应，数据库中还有**视图**（View）的概念。视图是从一个或几个基本表中导出的表。视图是一个虚表，即视图所对应的数据不实际存储在数据库中，数据库中只存储视图的定义。视图一经定义，可以和基本表一样被查询、被删除，也可以用来定义新的视图，但是数据更新（增、删、改）操作可能会受到一定的限制。

1.3.2 表之间的关系

一个关系数据库往往是由多个关系表组成的，有时为了尽量减少数据的冗余，需要将一个大的关系表分解为几个小的关系表。这些关系表是彼此联系地存在于数据库中的，两个关系表之间的联系可以分为三类：

1．一对一联系（1：1）

如果对于表 A 中的每一条记录，表 B 中至多有一条记录与之联系，反之亦然，则

称表 A 与表 B 具有一对一的联系，记为 1：1。

2．一对多联系（1：n）

如果对于表 A 中的每条记录，表 B 中有几条记录（n>0）与之联系；反之，对于表 B 中的每条记录，表 A 中至多有一个记录与之联系，则称表 A 与表 B 具有一对多的联系，记为 1：n。

3．多对多联系（m：n）

如果对于表 A 中的每条记录，表 B 中有 n 条记录（n>0）与之联系；反之，对于表 B 中的每条记录，表 A 中也有 m 个记录（m>0）与之联系，则称表 A 与表 B 具有多对多的联系，记为 m：n。

表之间的关系是通过外部码的联系来实现的，在 Access 2000 的学习过程中读者会清楚地认识到这一点。设定表之间的关系主要是为了实施表之间的参照完整性（Referential integrity）。比如这里有两个关系表：职工（ENO，ENAME，DNO）和部门（DNO，DNAME），职工表的主码为 ENO，部门表的主码是 DNO。在职工表中，DNO 是它的外部码，职工表中每条记录的 DNO 上的值只允许有两种可能：（1）取空值，表明这个职工尚未分配到某个部门；（2）若非空值，则 DNO 的值必须是部门表中某个记录中的 DNO 值，即此职工不可能分配到一个不存在的部门中，也就是说被参照的关系表部门中一定存在一个记录，它的主码值等于该职工表中的外部码值。这就是所谓的参照完整性。

1.4 数据库语言

用来生成数据库应用程序的语言主要分为三大类型。

1.4.1 过程化语言

标准的计算机程序设计语言如 Pascal、Basic 和 C 都是过程化语言，这些语言可以通过某种“应用程序接口（API）”来创建数据库应用程序。这种 API 由一组标准的函数（或调用）组成，这些函数和调用则扩展了语言的功能，使之能访问数据库中的数据。当程序设计人员用过程化语言创建数据库应用时，必须把应用的代码编写成一系列的过程，每个过程执行应用的某一部分的工作，如一个过程查询数据库，而另一过程更新数据库中的数据，然后不同的过程通过其他的用户界面过程（例如菜单系统）联系在一起，并且在应用中的适当地方运行。

上述这些过程化语言一般用来创建非数据库应用程序，它们通常被称为“第三代语言”（3GL）。还有一些过程化程序设计语言是某种特定的 DBMS 专用的， 这些语言一般被称为“第四代语言”（4GL），即数据库专用语言。常见的数据库专用的过程化语言如 DBASE 语言，Paradox 数据库的 PAL 语言等等。

需要全本请访问 www.ertongbook.com

关于过程化的语言读者可以在 FoxPro 的学习过程中使用到。

1.4.2 结构化查询语言（SQL）

结构化查询语言（Structure Query Language）是基于关系模型的数据库查询语言，它是一种非过程化的程序语言，也就是说，没有必要写出将如何做某事情，只需写出做到什么就可以了，具体的实现过程由数据库管理系统（DBMS）来实现。写出的 SQL 语句可看作是一个问题，称为“查询（Query）”，针对这个查询，就可以得到所需的查询结果。

SQL 已经成为了一种国际的标准，目前市场上各个数据库产品的各个厂家所推出的产品基本上都支持 SQL 语句。在利用 Access 2000 和 FoxPro 来管理数据的过程中，时常要和 SQL 语句打交道。SQL 的英语名称是结构化的查询语言，实际上它的功能包括查询、操纵、定义和控制四个方面，是一个综合的、通用的、功能极强的关系数据库语言。接下来我们将简要地介绍一下 SQL 语句的具体使用方法。

1．SQL 数据定义功能

可以利用 SQL 语句来定义基本表、定义视图和定义索引。其关键字为：

Create Table　Create View　Create Index　Drop Table
Drop View　Drop Index　Alter Table

- 利用 SQL 语句定义和修改基本表

定义基本表的语句格式为：

Create Table 表名（字段 1　类型，字段 2　类型……，字段 n　类型）；

比如：Create Table　学生（学号，char（3）　Not Null，
姓名，char（15），
系别，char（20），
年龄，SmallInt）；

执行此 Create Table 语句，就在数据库中建立了一个新的，空的学生表，其中包括四个字段。在“学号”字段的后面利用 Not Null 进行了修饰，规定了学号字段的取值不能为“空”，空值是指不知道或者不能用的值。

如果要在创建的学生表中增加一个字段，可以采用下面的 Alter Table 语句：

Alter Table　学生
ADD 性别 SmallInt；

要删除一个表，SQL 语句格式为：Drop Table 表名；在删除一个表时，将基本表的定义连同表上所有的记录、索引以及由此基本表导出的所有视图全部都删除，并释放相应的存储空间。

- 利用 SQL 语句建立和删除索引

对于一个基本表，可以根据应用环境的需要建立若干索引，以提供多种存取路径。建立索引的语句格式为：

Create Index　索引名
On 基本表名（字段 1，字段 2……）；

指定索引值的排列次序时，可以利用 ASC（升序）或 DESC（降序）来进行修饰，缺省值为升序。如果要求每一索引只对应惟一的数据记录，可以在索引名前面加上 Unique 进行修饰，比如我们给学生表创建惟一索引的语句如下：

Create Unique Index 学号索引　On 学生（学号 ASC）；

删除索引的语法为：Drop Index 索引名；在删除索引的同时，也会把有关索引的描述也从数据字典中删除。

- 利用 SQL 语句建立和删除视图

视图的创建和删除与表比较类似。

创建视图的语法格式为：

Create View 视图名（字段 1，字段 2……）
As 子查询；

其中涉及到查询语句，查询语句我们将接下来进行介绍，这里只是给出一个简单的例子，如：

Create View 男生（学号，姓名，年龄）
As Select 学号，姓名，年龄
From 学生
Where 性别=1；

在此视图中，将显示所有男生的信息（这里性别取值为 1 表示是男生）。

删除视图也要用到 Drop 语句，格式为：Drop View 视图名；在删除视图的同时，也将删除由此视图导出的其他视图。如果导出此视图的基本表被删除了，则此视图也将自动删除。

2．SQL 数据操纵功能

SQL 的数据操纵功能包括 Select，Insert，Delete 和 Update 四个语句，即检索和更新（包括增、删、改）两部分功能。

- SQL 查询语句（Select）

SQL 语言的核心是数据库查询语句，语句的一般格式为：

Select　目标字段
From　基本表（或者视图）
[Where　条件表达式]
[Group By　字段 1]
[Order By　字段 2]；

整个语句的含义是：根据 Where 子句中的条件表达式，从基本表（或视图）中找出满足条件的所有记录，按 Select 子句中的目标字段，选出记录中的分量形成结果表。如果有 Order 子句，则结果表要根据指定的字段 2 按升序或降序排序。Group 子句将结果按字段 1 分组，每个组产生结果表中的一个记录。SQL 语句对数据库的操作十分

灵活方便，下面我们举几个例子来说明其用法。

例 1　求经管学院学生的学号与姓名：

```
Select 学号，姓名
From 学生
Where 系别=‘经管学院’；
```

例 2　求所有有学生的系别：

```
Select Distinct 系别
From 学生；
```

例 3　求全体学生的所有信息：

```
Select * From 学生；
```

例 4　求年龄在 18~22 之间的学生的所有信息，并按照年龄降序排列：

```
Select *
From 学生
Where 年龄 Between 18 And 20
Order By 年龄 ASC；
```

例 5　求数学系和计算机系的学生：

```
Select *
From 学生
Where 系别 In （‘数学系’，‘计算机系’）；
```

例 6　求姓“肖”的所有学生：

```
Select *
From 学生
Where 姓名 Like ‘肖%’；
```

例 7　求所有年龄大于 20 的男生的信息：

```
Select *
From 学生
Where 年龄>20 And 性别=1；
```

例 8　求所有学生年龄的平均值：

```
Select AVG（年龄）
From 学生；
```

从上面的例子中可以看到，在 SQL 语句里，还可以使用到 DBMS 所提供的各种库函数，比如 Count（计数）、Sum（求和）、Avg（平均值）、Max（最大值）、Min（最小值）等。除了能进行简单的查询以外，还可以进行连接查询、嵌套查询等，下面我们再举几个稍微复杂的例子：

假设这里有两个相互关联的表：

学生（学号，姓名，年龄，性别，系别）；

成绩（学号，课号，成绩）；

学号是学生表的主关键字，在成绩表中以外部码的形式出现；成绩表中学号和课

号的组合作为主关键字，两个表之间是一对多的关系。

例 9　求学生以及他选修课程的课程号码和成绩：

```
Select 学生.*，成绩.*
From 学生，成绩
Where 学生.学号 = 成绩.学号；
```

例 10　求选修了“高等数学”课程（课号为 1234）的所有经管学院学生的姓名和成绩：

```
Select 学生.姓名，成绩.成绩
From 学生，成绩
Where 学生.系别 =‘经管学院’ And 成绩.课号 =‘1234’；
```

例 11　求修了课程号为 2345 的学生学号和姓名：

```
Select 学号，姓名
From 学生
Where 学生.学号 In
    Select 学号
    From 成绩
    Where 课号 =‘2345’；
```

在嵌套查询中，最常用的谓词是 In，也可以用量词 Exists 和 Not Exists 来执行嵌套查询，比如：

例 12　求修了课程号为 2345 的学生姓名：

```
Select 姓名
From 学生
Where Exists
    （Select *
    From 成绩
    Where 学号 = 学生.学号 And 课号 =‘2345’）；
```

在查询语句中还可以使用到 Union，Having 等子句，下面是一个使用 Union 的例子：

例 13　求经管学院的学生以及年龄小于 20 岁的学生：

```
Select *
From 学生
Where 系别 =‘经管学院’
Union
Select *
From 学生
Where 年龄<20；
```

通过嵌套查询和连接查询，可以组合出各种各样的查询语句，由于篇幅的限制，在这里不可能列出所有的 SQL 查询语句，读者可以在实际使用过程中自己去发掘。

3．SQL 更新语句

SQL 更新语句包括修改、删除和插入三类语句。

- 修改（Update）

修改语句的一般格式为：

```
Update 表名
Set 字段 = 表达式
[Where 条件子句]；
```

例 14　把数学系所有学生的成绩置为 0：

```
Update 成绩
Set 成绩 = 0
Where ‘数学系’=
        （Select 系别
        From 学生
        Where 学生.学号 = 成绩.学号）；
```

在 Update 语句中，一次只能对一个表更新，这样有可能破坏参照完整性，比如在学生表中将学生“张三”的学号改为“951000”，那么就会引起成绩表中张三的成绩的不一致性。解决的具体办法是同时执行另一个 Update 语句，将成绩表中张三的学号也换为“951000”。

- 删除（Delete）

删除语句的一般格式为：

```
Delete
From 表名
[Where 条件子句]；
```

例 15　删除经管学院、全体学生的成绩记录：

```
Delete
From 成绩
Where ‘经管学院’=
        （Select 系别
        From 学生
        Where 学生.学号 = 成绩.学号）；
```

- 插入（Insert）

插入语句的一般格式为：

```
Insert Into 表名（字段 1，字段 2……）
Values（常量 1，常量 2……）；
```

例 16　插入一个新学生：

```
Insert
Into 学生
```

Values（‘961234’，‘小刚’，20，1，‘材料工程系’）；

4．SQL 数据操纵功能

SQL 控制功能是指控制用户对数据的存取权力。某个用户对某类数据具有何种操作权力是由 DBA（数据库管理员）来决定的。授予用户操作权限以及撤消操作权限都可以用 SQL 语句来实现。

- 赋予用户操作权限

授权语句的格式为：Grant 权力 On 对象类型 对象 To 用户名；

比如把对成绩表查询的权力授予用户“用户 1”的 SQL 语句为：

Grant Select On Table 成绩 To 用户 1；

如果在授权语句后面加上 With Grant Option，表示获得这种权力的用户可以把权力再授予其他的用户。

- 撤消用户的操作权限

当某个用户不再存在时，需要撤消用户的操作权限，其 SQL 语句格式如下：

Revoke 权力 On 对象类型 对象 From 用户名；

比如撤消用户 1 对成绩表的查询操作，SQL 语句为：

Revoke Select On Table 成绩 From 用户 1；

上面我们只是对 SQL 语句的四类功能做了一个非常简要的介绍，SQL 语句非常灵活多样，SQL 语句写得好坏与否，有时直接关系到数据的检索、存取速度。完成同样的操作可以通过多种 SQL 语句来实现，用户可以从中选择一种最为喜爱的方法。在 Access 2000 中建立查询时可以直接利用标准的 SQL 语句来实现，在 FoxPro 中提供了一个命令窗口，用户在使用过程中会发现，利用 SQL 语句，可以非常直接、方便地实现建表、查询等操作。

在某些应用程序中，允许直接插入一些 SQL 语句（称为嵌入式 SQL）来完成数据的检索、更新等操作。读者如果有兴趣，可以查阅一些相关图书，在这里我们就不再介绍了。

1.4.3 其他语言

用于开发数据库应用程序的语言中，还可以使用目前数常见的“面向对象程序设计”（OOP）语言，如 C++、 Objact Pascal 等。OOP 代表了一种完全不同的程序设计方法， 在这种程序设计方法中，活动被定义为在“对象”上发生的操作，而不是作为一系列过程来定义的。在数据库应用程序中使用 OOP 语言的情况正在不断增加。

开发数据库应用程序使用的另一种语言是“宏”语言。宏语言不是一种完全的程序设计语言，它实际上是一个用户手工输入的表，这个表被输入到应用程序中，以便自动执行一定的任务。对于某个特定应用的高级语言，宏语言通常可以在低档 DBMS 软件中或数据库服务器的前端中找到。

需要全本请访问 www.ertongbook.com