

中文

Access

2000

使用指南

廖疆星 林涛 肖金秀 编著

简

简单的解决方案

高

高级的使用技巧

详

详尽的培训教材

冶金工业出版社

中文 Access 2000 使用指南

廖疆星 林 涛 肖金秀 编著

冶金工业出版社

2000 · 北京

内容简介

Access 2000 是微软公司最新推出的关系型数据库管理系统, 本书全面深入地介绍了 Access 2000 的基本知识和高级使用方法, 包括: 数据库的基础知识、Access 数据库的规划构造、表的设计与使用、数据查询及高级使用方法、窗体及控件的使用、报表的创建与高级使用技巧、VBA 语言的应用、数据库的 Web 发布、数据库的维护与管理、创建应用程序等。

本书编写, 力求做到语言描述形象化, 理论、概念简单化, 同时强调可操作性和实用性, 以便读者真正掌握这个功能强大的数据库开发工具。

图书在版编目 (CIP) 数据

中文 Access 2000 使用指南 / 廖疆星等编. -北京:
冶金工业出版社, 2000.8
ISBN 7-5024-2606-X

I. 中... II. 廖... III. 关系数据库-数据库管理
系统, Access 2000-指南 IV. TP311.138-62

中国版本图书馆 CIP 数据核字 (2000) 第 28429 号

中文 Access 2000 使用指南

廖疆星 林涛 肖金秀 编著

出版 冶金工业出版社
社址 北京沙滩嵩祝院北巷 39 号, 邮编 100009
发行 冶金工业出版社发行; 各地新华书店经销
印刷 广东出版技校彩印厂印刷
开本 787×1092 毫米 1/16
印张 19.5
字数 523 千字
版本 2000 年 8 月第 1 版 2000 年 8 月第 1 次印刷
印数 1-1900 册

书号 ISBN 7-5024-2606-X / TP·175

定价 30.00 元

前 言

Access 2000 是微软公司最新推出的关系型数据库管理系统，它不但易学、易用、功能强大，而且与其它应用程序高度集成，具有强大的网络功能，又提供了大量的管理工具和许多简便有效的向导，使用户无需花费大量的时间和精力去学习数据库知识即可开发出高效实用的数据库应用系统。

Access 2000 增加了很多新的特性，主要表现在与 Office 套件其它应用程序的集成、与 Web 的组合更紧密以及向企业级数据库的进一步延伸。如首次将数据库转换成以前的 Access 版本、新的数据库窗口外观、对象名称自动更正、子数据表、数据访问页等。

本书介绍 Microsoft Access 2000 的基本知识，包括：数据库的基础知识、Access 2000 数据库的基本操作、表的设计与使用、数据库中的数据查询、创建窗体、使用控件、报表的创建与使用、图表的应用、编程工具的使用等。

本书使用更多的篇幅介绍 Microsoft Access 2000 的高级知识，包括：

- Access 数据库的规划构造：数据库设计原则、数据表设计的方法等。
- 表中数据的操作：特殊类型字段的编辑、排序记录、筛选记录等。
- 查询的高级使用方法：深入 SQL 查询、多个表的联接、查询准则详述、执行计算的方法、查询的属性设置、优化查询的方法等。
- 高级报表技巧：为选定的记录生成报表、子报表、多列报表、报表快照、优化报表的性能等。
- VBA 语言的应用：VBA 代码的编写、VBA 语言中的过程、数据类型、常数变量和数组、语句、对象属性方法和事件等。
- 数据库的 Web 发布：创建和编辑数据访问页、Web 发布数据库等。
- 数据库的维护与管理：数据库安全、以 MDE 文件保护数据库、优化数据库性能等。
- 交换数据：导入数据、导出数据、创建数据链接等。
- 定制 Access 2000：“选项”对话框参数的设置、自定义菜单和快捷菜单、自定义工具栏等。
- 创建应用程序：切换面板、启动界面的设置等。
- 使用项目：创建 Access 项目、连接 Microsoft SQL Server 数据库等。

这些内容深入探讨了 Access 2000 的一些高级实用技巧。

最后提供了两个有完整开发过程的实例，读者可根据自己学习的不同阶段予以选读。

为使读者在最短时间内掌握开发 Access 数据系统的基本方法，以及开发功能更强、使用更方便的数据库系统所需要用到的高级技巧，本书在编写过程中，对一些重要知识进行了深入的探讨，而且插入了大量图片和实例，力求做到语言描述形象化，理论、概念简单化，同时强调可操作性 and 实用性，以便读者真正迅速掌握这个功能强大的数据库开发工具。

由于编者水平有限，时间仓促，缺点错误在所难免，恳请读者批评指正。

编 者
2000 年 6 月

目 录

第一章 数据库的基础知识 1	2.6.3 管理数据库对象和组..... 23
1.1 数据库技术的发展..... 1	2.7 新建一个数据库..... 23
1.1.1 人工管理阶段..... 1	2.7.1 新建一个空的数据库..... 24
1.1.2 文件系统阶段..... 2	2.7.2 利用向导创建一个数据库..... 24
1.1.3 数据库系统阶段..... 2	2.8 获取 Access 2000 的帮助信息..... 27
1.2 数据库管理系统..... 3	2.8.1 使用 Office 助手..... 27
1.3 关系数据库..... 4	2.8.2 使用 Microsoft Access 帮助窗口..... 27
1.3.1 关系模型..... 5	综合练习二..... 28
1.3.2 关系数据库 DML 的特点..... 5	一、基础题..... 28
1.4 关系数据库标准语言——SQL..... 6	二、上机操作..... 28
1.4.1 SQL 数据库的体系结构..... 6	第三章 Access 数据库的规划构造 29
1.4.2 SQL 数据定义..... 6	3.1 总体设计..... 29
1.4.3 SQL 数据操纵功能..... 7	3.1.1 需求分析..... 29
1.4.4 SQL 数据控制功能..... 8	3.1.2 数据库设计原则..... 30
1.4.5 视图..... 8	3.2 详细设计..... 30
1.5 数据库设计..... 8	3.2.1 数据表的设计..... 30
1.6 数据库保护..... 10	3.2.2 创建窗体和报表及其他数据库对象..... 31
1.6.1 数据库的安全性..... 10	3.2.3 输入数据..... 33
1.6.2 数据库的完整性控制..... 10	3.3 优化设计..... 33
1.6.3 数据库的并发控制..... 11	3.3.1 表分析器向导..... 33
1.6.4 数据库的恢复..... 11	3.3.2 性能分析器..... 35
1.7 Access 2000 和数据库的关系..... 12	综合练习三..... 36
综合练习一..... 12	一、基础题..... 36
一、基础题..... 12	二、上机操作..... 36
二、上机操作..... 12	第四章 表的应用 37
第二章 Access 2000 数据库的基本操作 13	4.1 使用设计器创建一个新表..... 37
2.1 Microsoft Access 概述..... 13	4.2 使用向导创建一个新表..... 40
2.2 Access 2000 的新特点..... 13	4.3 通过输入数据创建一个新表..... 43
2.3 启动 Access 2000..... 14	4.4 编辑表中的数据..... 43
2.4 Access 2000 的工作界面..... 16	4.4.1 增加记录..... 43
2.5 数据库对象..... 17	4.4.2 删除记录..... 44
2.5.1 表..... 17	4.4.3 将表中的数据排序..... 44
2.5.2 查询..... 18	4.4.4 数据的筛选..... 45
2.5.3 窗体..... 19	4.4.5 数据的查找..... 45
2.5.4 报表..... 20	4.5 设置表之间的关系..... 45
2.5.5 页..... 20	4.6 子表的使用..... 47
2.5.6 宏..... 21	4.7 浏览、编辑记录..... 48
2.5.7 模块..... 21	4.7.1 在数据表视图中浏览记录..... 48
2.6 数据库窗口的使用..... 22	4.7.2 编辑记录..... 50
2.6.1 设置数据库对象的属性..... 22	4.8 编辑特殊类型的字段..... 51
2.6.2 更改数据库对象的显示方式..... 22	4.8.1 日期/时间..... 51

目 录

4.8.2 备注	51	5.5.1 添加或删除表	87
4.8.3 是/否	52	5.5.2 修改字段	88
4.8.4 超级链接	52	5.5.3 更改设计网格的显示	89
4.8.5 OLE 对象	53	5.6 高级查询	90
4.9 查找与替换记录	54	5.6.1 查询中的计算	90
4.9.1 查找数据	54	5.6.2 使用多个数据表	95
4.9.2 替换记录内容	55	5.6.3 联接的相关操作	96
4.9.3 使用通配符	56	5.6.4 查询准则详述	100
4.10 排序记录	56	5.6.5 深入 SQL 查询	105
4.11 筛选记录	57	5.6.6 优化查询的方法	108
4.11.1 按选定内容筛选	57	综合练习五	109
4.11.2 按窗体筛选	58	一、基础题	109
4.11.3 输入筛选目标	59	二、上机操作	109
4.11.4 高级筛选/排序	59	第六章 创建窗体	111
4.12 定制数据表视图	59	6.1 窗体的基本结构与类型	111
4.12.1 改变行高和列宽	60	6.1.1 窗体的基本结构	111
4.12.2 移动列	60	6.1.2 窗体的基本类型	112
4.12.3 冻结列	60	6.2 利用窗体向导创建窗体	113
4.12.4 隐藏列	61	6.2.1 自动创建窗体	114
综合练习四	61	6.2.2 利用窗体向导创建简单窗体	115
一、基础题	61	6.2.3 利用窗体向导创建主/子式窗体	116
二、上机操作	62	6.2.4 利用窗体向导创建数据透视表式窗体	118
第五章 使用查询	63	6.2.5 利用窗体向导创建图表式窗体	121
5.1 概述	63	6.3 窗体格式的详细设计	123
5.2 查询类型	63	6.3.1 窗体的“设计”视图	123
5.2.1 选择查询	63	6.3.2 窗体格式的一些设置	123
5.2.2 操作查询	64	综合练习六	125
5.2.3 交叉表查询	68	一、基础题	125
5.2.4 参数查询	69	二、上机操作	125
5.2.5 SQL 查询	69	第七章 控件的使用	126
5.3 查询的属性	70	7.1 控件的种类	126
5.3.1 操作查询的权限	70	7.1.1 控件的基本类型	126
5.3.2 设置查询的属性	71	7.1.2 用于输入、显示和筛选数据的控件	127
5.3.3 设置字段的属性	75	7.2 常用控件的创建	129
5.3.4 设置字段列表的属性	78	7.2.1 创建命令按钮	129
5.3.5 字段的格式属性	78	7.2.2 创建组合框	130
5.4 用查询向导创建查询	81	7.2.3 创建选项组	131
5.4.1 简单查询向导	82	7.2.4 创建子窗体/子报表	133
5.4.2 交叉表查询向导	83	7.3 控件的编辑	134
5.4.3 查找重复项查询向导	84	7.4 控件的属性	135
5.4.4 查找不匹配项查询向导	86		
5.5 在设计视图中修改查询	87		

目 录

7.5 控件的事件生成器	136	10.2.2 创建宏组	175
综合练习七	138	10.2.3 执行和调试宏	176
一、基础题	138	10.3 Visual Basic for Applications	177
二、上机操作	138	10.3.1 模块简介	177
第八章 设计报表	139	10.3.2 VBA 语言中的过程	178
8.1 了解报表	139	10.3.3 数据类型、常数、变量和数组	186
8.1.1 表格式报表	139	10.3.4 运算符	194
8.1.2 纵栏式报表	139	10.3.5 语句	200
8.1.3 标签	140	10.3.6 对象、属性、方法和事件	205
8.2 快速创建报表	140	10.3.7 创建和运行自定义的函数和过程	208
8.3 利用向导创建报表	141	10.3.8 关于运行 Visual Basic 代码	209
8.3.1 创建表格式报表	141	10.4 编程工具的选择	210
8.3.2 创建标签	144	综合练习十	211
8.4 报表的设计视图	145	一、基础题	211
8.4.1 报表的几个组成部分	145	二、上机操作	211
8.4.2 使用控件	146	第十一章 数据库的 Web 发布	213
8.5 高级报表技巧	148	11.1 Microsoft Access 2000 中的 Web 特征	213
8.5.1 为选定的记录生成报表	148	11.1.1 Web 介绍	213
8.5.2 子报表	151	11.1.2 Access 中的超级链接	213
8.5.3 多列报表	154	11.1.3 数据访问页	214
8.5.4 在报表中显示输入的参数	155	11.2 创建数据访问页	215
8.5.5 报表快照	156	11.2.1 自动创建数据访问页	215
8.5.6 优化报表的性能	157	11.2.2 运用向导创建数据访问页	215
8.6 报表的页面设置和打印	159	11.2.3 利用现有的数据访问页创建新的数据访问页	217
综合练习八	160	11.3 编辑数据访问页	217
一、基础题	160	11.3.1 设计视图	217
二、上机操作	160	11.3.2 HTML 语句编辑器	218
第九章 图表的应用	161	11.4 在 Web 中发布和访问 Access 数据库	218
9.1 在数据访问页中创建图表	161	综合练习十一	220
9.2 编辑图表	163	一、基础题	220
9.2.1 编辑窗体或报表中的图表	163	二、上机操作	220
9.2.2 编辑数据访问页中的图表	168	第十二章 数据库的维护与管理	222
9.3 将图表链接到记录	170	12.1 数据库安全	222
综合练习九	171	12.1.1 设置打开数据库的密码	222
一、基础题	171	12.1.2 用户级的安全机制	223
二、上机操作	171	12.1.3 设置安全机制向导	225
第十章 编程工具	172	12.2 生成 MDE 文件	227
10.1 事件	172	12.3 优化数据库性能	228
10.2 宏	173	12.3.1 压缩数据库	228
10.2.1 宏的基本组成部分	174	12.3.2 性能分析器	228

目 录

综合练习十二.....	229	15.2.2 使用“Autoexec”宏.....	257
一、基础题.....	229	综合练习十五.....	259
二、上机操作.....	229	一、基础题.....	259
第十三章 交换数据.....	230	二、上机操作.....	259
13.1 可以存取的数据格式.....	230	第十六章 使用项目.....	260
13.2 导入数据.....	230	16.1 了解 Access 项目.....	260
13.3 导出数据.....	232	16.1.1 Access 项目是什么.....	260
13.4 创建数据链接.....	233	16.1.2 MSDE.....	261
13.5 使用 Word 合并数据.....	235	16.2 Access 项目与 Access 数据库的比较.....	264
13.6 使用 Word 发布数据.....	237	16.3 创建 Access 项目.....	266
13.7 使用 Excel 分析数据.....	237	16.4 连接 Microsoft SQL Server 数据库.....	267
13.8 发送电子邮件.....	238	16.5 “罗斯文商贸”项目.....	268
综合练习十三.....	239	16.5.1 安装罗斯文商贸项目.....	268
一、基础题.....	239	16.5.2 打开罗斯文商贸项目.....	268
二、上机操作.....	239	综合练习十六.....	269
第十四章 定制 Access 2000.....	240	一、基础题.....	269
14.1 “选项”对话框.....	240	二、上机操作.....	269
14.2 自定义菜单.....	242	第十七章 实例与技巧.....	270
14.3 自定义快捷菜单.....	245	17.1 收文登记数据库开发与技巧.....	270
14.4 自定义工具栏.....	248	17.1.1 收文登记数据库的需求及总体规划.....	270
14.5 自定义启动后的界面.....	250	17.1.2 表的设计.....	270
综合练习十四.....	250	17.1.3 查询的设计.....	276
一、基础题.....	250	17.1.4 报表的设计.....	277
二、上机操作.....	251	17.1.5 窗体的设计.....	279
第十五章 创建应用程序.....	252	17.1.6 数据库安全设置.....	284
15.1 切换面板.....	252	17.2 进销存管理系统开发与技巧.....	287
15.1.1 “联系人”数据库中的切换面板.....	252	17.2.1 需求分析.....	287
15.1.2 启动“切换面板管理器”.....	254	17.2.2 表的设计.....	288
15.1.3 增加切换面板页.....	254	17.2.3 查询的设计.....	293
15.1.4 主切换面板中的项目.....	255	17.2.4 窗体的设计.....	296
15.1.5 切换面板中的图像.....	256	17.2.5 安全机制.....	300
15.2 使启动界面更友好.....	257		
15.2.1 使用“启动”选项.....	257		

第一章 数据库的基础知识

本章重点

- 数据库管理系统
- 关系数据库
- SQL

1.1 数据库技术的发展

人们对计算机的应用从科学计算进入数据处理是一个划时代的转折，使计算机从单一的科学计算机工具变为群众日常工作中的得力助手。为了借助计算机科学地保存和管理复杂的大量数据，以便人们能方便而充分地利用这些宝贵的信息资源，经过不断的发展，逐渐形成了数据库技术。

数据库 (Database, 简称 DB) 技术是计算机软件的一个重要分支，产生于 60 年代末。1969 年美国 IBM 公司推出了层次模型的 IMS 数据库系统，1969 年 10 月，提出 COBOL 语言的美国 CODASYL 组织的数据库任务组 (DBTG) 发表了网状数据库系统的标准文本，1970 年美国 IBM 公司的高级研究员 E.F.Codd 提出了关系模型。这三件事奠定了数据库系统的基础。70 年代、80 年代，数据库技术在理论上和实践中得到飞速发展，逐步完善，并不断向更高层次发展。

数据库技术的发展大致经过以下三个阶段。

- 1) 人工管理阶段。
- 2) 文件系统阶段。
- 3) 数据库系统阶段。

1.1.1 人工管理阶段

这一阶段 (20 世纪 50 年代以前) 计算机应用的背景是：计算机主要用于科学计算，从硬件上看，外存只有磁带、卡片、纸带，没有磁盘等直接存取的存储设备；从软件上看，没有操作系统，没有管理数据的软件，数据处理的方式是批处理。

这个时期数据管理的特点是：

1) 数据不保存。因为计算机主要应用于科学计算，一般不需要将数据长期保存。只是在计算某一课题时将数据输入，用完就撤走，不仅对用户数据这样处理，有时对系统软件也是这样。

2) 没有专用的软件对数据进行管理。程序员不仅要规定数据的逻辑结构，而且还要在程序中设计物理结构，包括存储结构，存取方法，输入输出方式等。因此程序中存取数据的子程序随着存储的改变而改变。即数据与程序不具有独立性，这样不仅程序必须花费许多精力在数据的物理布置上，而且数据在存储上有一点改变，就必须修改程序。

3) 只有程序 (Program) 概念，没有文件 (File) 概念。数据的组织方式必须由程序员自行设计。

4) 一组数据对应一个程序，数据是面向应用的。即使两个应用程序涉及某些相同的数据，也必须各自定义，无法互相利用，互相参照。所以程序和程序之间有大量重复的数据。

以上特点可用图形来表示,如图 1-1 所示。

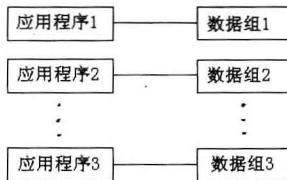


图 1-1

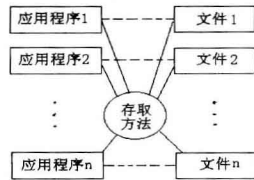


图 1-2

1.1.2 文件系统阶段

这一阶段(20世纪50年代后期到60年代中期)应用计算机的情况是:计算机不仅用于科学计算机,还大量用于管理。外存储器有了磁盘、磁鼓等直接存取的存储设备。在软件方面,操作系统中已经有了专门的管理数据软件,一般称为文件系统(有时称为“信息处理模块”)。从处理方式讲不仅有了文件批处理,而且能够联机实时处理。

这一阶段的数据管理形成了如下几个特点:

1) 因为计算机大量用于数据处理,数据需要长期保留在外存上反复处理,即经常需要对文件进行查询、修改、插入和删除等操作。

2) 有了软件进行数据管理,程序和数据之间有软件提供存取方法进行转换,有共同的数据查询修改的管理模块。文件的逻辑结构与存储结构由系统进行转换,使程序与数据有了一定的独立性。这样程序员可以集中精力于算法,而不必过多地考虑物理细节。并且,数据在存储上的改变不一定需要改变程序,大大节省维护程序的工作量。

3) 文件组织已多样化。有索引文件、链接文件和直接存取文件等。文件之间是独立的,联系要通过程序去构造。

4) 数据不再属于某个特定的程序,可以重复使用。但程序仍然基于特定的物理结构和存取方法,因此数据结构与程序之间的依赖关系并未根本改变。

5) 上述特点比人工管理阶段有很大的改进,但随着数据量的急剧增加,数据管理规模的扩大,文件系统显露出三个缺点:

① 数据冗余度(Redundancy)大。这是由于文件之间缺乏联系,造成每个应用程序都有对应的文件,有可能同样的数据在多个文件中重复存储。

② 数据不一致性。这是由数据冗余造成的,稍不谨慎,就可能同样的数据在不同的文件中不一样。

③ 数据和程序缺乏独立性。文件系统中文件是为某一特定应用服务的。文件的逻辑结构对该应用程序来说是优化的。因此,要想对现有的数据再增加一些新的应用是很困难的,系统不容易扩充。一旦数据的逻辑结构改变,就必须修改应用程序,修改文件结构的定义。而应用程序的改变,如应用程序所使用的高级语言的变化等,也将影响文件的数据结构的改变。数据和程序缺乏独立性。这个时期程序与数据的关系如图 1-2 所示。

1.1.3 数据库系统阶段

这一时期(20世纪60年代后期开始)计算机应用越来越广泛,数据量急剧增长,而且数据的共享要求越来越高。这时,有了大容量的磁盘,联机实时处理要求更多了,并开始提出和考虑分布处理。另外,软件价格开始上升,硬件价格下降,使编制和维护系统软件及应

用程序所需的成本相对增加。在这种情况下,为了解决多用户、多应用共享数据的需求,使数据为尽可能多的应用服务,出现了数据库这样的数据管理技术。特点如下:

1. 采用复杂的数据模型(结构)

数据模型不仅描述数据本身的特点,还要描述数据之间的联系。这种联系通过存取路径实现。通过所有存取路径表示自然的数据联系是数据库与传统文件的根本区别。这样数据不再面向特定的某个或多个应用,而是面向整个应用系统。数据冗余明显减少,实现了数据共享。

2. 有较高的数据独立性

数据的物理结构与逻辑结构间差别可以很大。用户以简单的逻辑结构操作数据而无需考虑数据的物理结构。数据库的结构分成用户的逻辑结构、整体逻辑结构、物理结构三级。用户的数据和外存中的数据之间转换由数据管理系统实现。在物理结构改变时,尽量不影响整体逻辑结构、用户的逻辑结构以及应用程序,这就是物理数据独立性。在整体逻辑结构改变时,尽量不影响用户的逻辑结构以及应用程序,这是逻辑数据独立性。

3. 提供方便的用户接口

数据库系统为用户提供了方便的用户接口,用户可使用查询语言或简单的终端命令操作数据库,也可以用程序方式操作数据库。

4. 提供数据控制功能

数据库管理系统提供以下四方面的数据控制功能:

- 1) 数据完整性。保证数据库始终包含正确的数据。用户可以设计一些完整性规则以确保数据值的正确性。
- 2) 数据安全性。保证数据的安全和机密,防止数据丢失或被窃取。
- 3) 数据库的并发控制。避免并发程序之间的相互干扰,防止数据库被破坏,杜绝提供给用户不正确的数据。
- 4) 数据库的恢复。在数据库被破坏或数据不可靠时,系统有能力把数据库恢复到最近某时刻的正确状态。

这一阶段程序和数据的关系如图 1-3 所示。

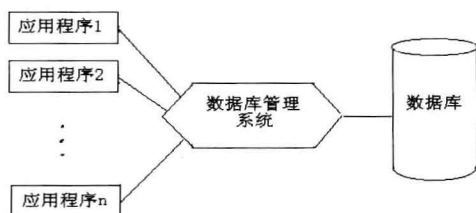


图 1-3

综上所述,可以说数据库是个通用的综合性的数据集合,它可以供各种用户共享且具有最小的冗余度和较高的数据与程序的独立性。

由于多种程序并发地使用数据库,为了能有效、及时地处理数据,并提供安全性和完整性,必须有一个软件系统在建立、运用和维护时对数据库进行统一控制,这种软件系统称为数据库管理系统。

1.2 数据库管理系统

要了解数据库管理系统(Data Base Management System,简称 DBMS),先要了解数据

库系统（简称 DBS）。

数据库系统是一个复杂的系统，它是采用了数据库技术的计算机系统。是一个实际可运行的，按照数据库方式存储、维护和向应用系统提供数据或信息支持的系统。它是存储介质、处理对象和管理系统的集合体，通常由数据库、硬件、软件和数据管理员四部份组成。

数据库管理系统是指在 DBS 中对数据进行管理的软件系统，它是 DBS 的核心组成部份。DBS 中所有与数据库打交道的操作，包括建库、查询、更新及各种数据控制，都是通过 DBMS 进行的。

用户要对数据库进行操作，是由 DBMS 把操作从应用程序带到外部级、概念级，再导向内部级，进而操纵存储器中的数据。一个 DBMS 的主要目标是把数据作为可管理的资源来处理。DBMS 应使数据易于为各种不同用户共享，应该增进数据的安全性、完整性和可用性，并提供高度的数据独立性。

DBMS 的主要功能包括以下几个方面：

1. 数据库的定义功能

DBMS 提供数据定义语言（DDL）定义数据库的结构，包括外模式、概念模式、内模式及其相互之间的映象，定义数据的完整性约束、保密限制等条件。因此在 DBMS 中包括 DDL 的编译程序。

2. 数据库存取功能

DBMS 提供数据操纵语言（Data Manipulation Language，简称 DML）实现对数据库数据的基本操作：检索、插入、修改和删除。

DML 有两类：一类是嵌入主语言中的，如嵌入 COBOL，FORTRAN，C 等高级语言中。这类 DML 语言本身不能独立使用，因此称为宿主型的。另一类是交互式命令语言，它们语法简单，可能独立使用，所以称为自主型或自含型的。DBMS 中必须包括 DML 的编译程序或解释程序。

3. 数据库运行控制功能

DBMS 对数据库的控制主要能过四个方面实现：数据库的恢复、数据完整性控制、多用户环境下的并发控制和数据安全性控制。因此 DBMS 中应该包括这四个子系统。

1) 数据库恢复。在数据库被破坏，或数据变得不正确时，系统有能力把数据库恢复到最近某个正确的状态。

2) 数据完整性控制。保持进入数据库中的数据语言的正确性和有效性，防止任何对数据造成错误的操作。

3) 并发控制。多个用户同时对数据库操作可能会破坏数据的正确性；或数据库存储了错误的数据库，或用户读了不正确的数据库。并发控制机构能防止上述情况发生，正确处理多用户、多任务环境下的并发操作。

4) 数据安全性控制。作用是防止未被授权的用户蓄谋或无意存取数据库中的数据。

4. 数据库的维护功能

包括数据库的初始数据载入、转换功能、转储功能、数据库的改组，以及性能监视分析功能等。

DBMS 的功能随不同的系统而有所差异，通常大型系统功能较强、较全，小型系统功能较弱。

1.3 关系数据库

按所基于的数据模型的不同，常见的 DBMS 可分为网状数据库、层次数据库和关系数

据库等。

其中最常见的是关系数据库，Access 2000 就是典型的关系数据库。下面简单介绍关系数据库的一些基本知识。

1.3.1 关系模型

关系数据库所基于的数据模型称为关系模型，是用二维表格结构表示实体类型，关键码表示实体间联系的数据模型。

二维表格是指各种数据以不同表格方式存储，各表格之间以关键字段相关联，构成一定的关系。如图 1-4 所示的表格，存储一些人事信息。

姓名	性别	年龄	职位	年薪
林勇	男	24	总经理	100000
王祥	男	25	开发部经理	100000
徐小丽	女	22	公关部经理	70000

图 1-4

在关系模型中，字段称为属性（attribute），字段值称为属性值，记录类型称为关系模式，记录称为元组（tuple），元组的集合称为关系（relation）或实例（instance）。有时，会直接称呼表格的元组为行，属性为列。在一个关系中，能唯一标识元组的属性集称为关系的候选键，其中，被选用的候选键称为关系的主键。

对于一个关系，应具备以下特点：

- 1) 不允许有两行完全相同（即表格中没有完全相同的记录）。
- 2) 用户不须考虑行序和列序。
- 3) 每一个属性值是基本的、不可分裂的。

为了维护数据库中数据与现实世界的一致性，关系数据库的插入、删除和修改操作必须遵循下述三类完整性规则：

- 1) 实体完整性规则。要求关系中元组的主键值不能是空值。
- 2) 引用完整性规则（reference integrity rule）。要求不能引用不存在的记录。
- 3) 用户定义的完整性规则。这是针对某一具体数据的约束条件，由实际应用环境决定。

反映某一具体应用所涉及的数据必须满足的语义要求。关系模型必须提供定义和检验这类完整性的机制，以使用统一的、系统的方法处理它们，不应由应用程序行使这一功能。

1.3.2 关系数据库 DML 的特点

关系数据库的 DML 语句分为查询语句和非查询语句两类。查询语句指的是用户要进行的各种检索操作，非查询语句指的是用户要进行的插入、删除、修改等操作。

关系数据库 DML 具有以下特点：

1. 使用上的方便性

关系模型只要程序员和用户了解关系的概念就可以了。关系模型具有关系运算的理论基础，这是一个很科学而成熟的理论。关系数据库语言适用于各类计算机用户，这些优点，使关系数据库逐渐取代了网状数据库和层次数据库。

2. 实现的难度

由于关系数据库中，关系以文件形式存储，数据间的联系主要靠 DML 实现，造成效率较低。现时计算机硬件得到飞速发展，关系数据库的效率已接近甚至超过网状数据库、层次

数据库。

3. 查询的非过程性

过程性指程序编制时需说明某个访问应该“如何做”的计算机语言。COBOL 一类高级语言、网状数据库和层次数据库的 DML 都属于过程性语言。而关系数据库 DML 对数据库的某个操作，一般只需说明“做什么”就可以了，具体实现由系统完成，因此称关系模型的 DML 是非过程性的。

1.4 关系数据库标准语言——SQL

SQL (Structured Query Language, 结构化查询语言) 是 1974 年由 Boyce 和 Chamberlin 提出的, 并在 IBM 公司的关系 DBMS SYSTEM R 上实现。它的前身是 1972 年提出的 SQUARE (Specifying Queries As Relational Expression) 语言, 在 1974 年作了修改, 改名为 SEQUEL (Structured English Query Language) 语言。后简称为 SQL。

SQL 是集 DDL, DML, 数据控制功能于一体的数据库语言。SQL 的 DML 是介于于关系代数和关系演算之间的一种语言。由于 SQL 使用方便、功能丰富、语言简洁易学, 很快得到推广和应用。

1986 年 10 月, 美国国家标准局 (ANSI) 的数据库委员会 X3H2 批准了 SQL 作为关系数据库语言的美国标准。同年公布了标准 SQL 文本。此后不久, 国际化标准组织 (ISO) 也作出了同样的决定, 后经修订, 在 1987 年 4 月颁了增强完整性特征的 SQL 89 版本。我国也制订了国家标准, 类似于 SQL 89 版本。

SQL 成为国际标准以后, 数据库产品的各个厂家纷纷推出各自的支持 SQL 的软件或者与 SQL 的接口软件。

这样, 不管是微机、小型机和大型机的数据库系统都采用 SQL 作为共同的数据存取语言和标准接口, 从而使未来的数据库世界有可能连接为一个统一的整体。

1.4.1 SQL 数据库的体系结构

基本上, 基于 SQL 语言的数据库的体系结构是三级结构, 如图 1-5 所示。

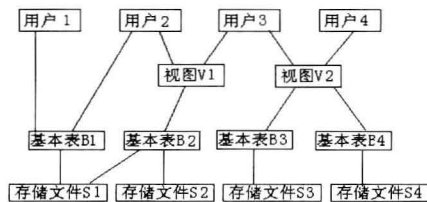


图 1-5

用户可以用 SQL 语言对视图和基本表进行查询等操作, 在用户来看, 视图和基本表是一样的, 都是关系。

视图是从一个或几个基本表导出的表, 本身不独立存储在数据库中。基本表是本身独立存在的表。

1.4.2 SQL 数据定义

SQL 的数据定义功能包括三部分, 定义基本表、视图和索引, 如下:

CREATE TABLE CREATE VIEW CREATE INDEX

DROP TABLE DROP VIEW DROP INDEX

ALTER TABLE

1. 基本表的定义

CREATE TABLE 表名 (列名1 类型[NOT NULL]
[, 列名2 类型 [NOT NULL]] ...)
[其他参数];

2. 修改基本表定义

ALTER TABLE 表名
ADD 列名 类型;

3. 删除基本表

DROP TABLE 表名;

4. 索引的建立和删除

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。语句格式为:

CREATE [UNIQUE] INDEX 索引名
ON 基本表名 (列名[次序][, 列名[次序]]...)
[其他参数];

1.4.3 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(增、删、改)。

1. 查询语句

数据库查询语句是 SQL 语言的核心,语句的一般格式是:

SELECT 目标列
FROM 基本表(或视图)
[WHERE 条件表达式]
[GROUP BY 列名1[HAVING 内部函数表达式]]
[ORDER BY 列名2[ASC/DESC]];

2. 更新语句

SQL 的更新语句包括修改、删除和插入三类语句:

1) 修改语句的格式为:

UPDATE 表名
SET 字段=表达式[, 字段=表达式]...
[WHERE 谓词];

2) 删除语句的格式为:

DELETE
FROM 表名
[WHERE 谓词];

3) 插入语句的格式为:

INSERT
INTO 表名[(字段名[, 字段名]...)]
VALUES (常量[, 常量]...);

1.4.4 SQL 数据控制功能

SQL 数据控制功能指控制数据库中用户的存取能力，具有如下功能：

- 1) 把授权的情况通知系统。由 GRANT 和 REVOKE 语句完成。
- 2) 将授权结果存入数据字典。
- 3) 对用户提出的操作请求，根据授权情况进行检查，以决定执行还是拒绝执行。

授权语句的一般格式为：

```
GRANT 权力[, 权力]...[ON 对象类型 对象名] TO 用户[, 用户]...;  
[WITH GRANT OPTION];
```

对不同类型的对象有不同的操作权力。

- 1) 基本表、视图及其字段：操作权力有查询、插入、更新、删除。
- 2) 基本表：操作权力还有修改表和建立索引。
- 3) 数据库：操作权力有建立表。
- 4) 表空间：使用数据库空间存储基本表的权力。
- 5) 系统权力：建立新数据库。

1.4.5 视图

视图是从一个或几个基本表导出的表。具有权限的用户可能定义若干视图。视图和基本表不同，视图是一个虚表，视图所对应的数据实际上并不存储在数据库中，只存储视图的定义。视图可以和基本表一样被查询、删除，并且可以用来定义新的视图。

SQL 建立视图的语句格式为：

```
CREATE VIEW 视图名[(字段名[, 字段名]...)]  
AS 子查询  
[WITH CHECK OPTION];
```

对视图进行查询时，系统首先把它转换成相同效果的对基本表的查询，然后执行修改了的查询，因此，基本表的变化可以反映到视图上，通过视图可以看到基本表动态的变化。

视图具有很多优点，主要有：

- 1) 对于数据库的重新构造提供一定程序的逻辑独立性。

逻辑独立性是相对数据的物理独立性而言的。数据的物理独立性指：用户和用户程序不依赖数据库的物理结构。数据的逻辑独立性是指当数据库重构造时，用户和用户程序不受影响。

- 2) 简化用户感觉。

视图的机制使用户将注意力集中于数据上，若数据不是直接来自基本表，则可以定义视图，从而使用户看到的数据简单直接，大大简化用户的数据查询操作。

- 3) 使不同的用户以不同的方式看待同一数据。
- 4) 对机密数据提供自动的安全保护功能。

运用视图，可以把机密数据从公共的数据视图中分离出去，即针对不同用户定义不同的视图，在用户视图中不包括机密数据的字段，使这类数据不能经由视图被用户存取。

1.5 数据库设计

现时，数据库已用于各类应用系统，如办公系统、管理信息系统、决策支持系统等，已成为现代信息系统的基础与核心部分。数据库设计的好坏直接影响整个数据库系统的效率和

质量。数据库设计是研制数据库及其应用系统的技术，是数据库在应用领域中主要的研究课题。数据库设计是指对于一个给定的应用环境，构造最优的数据库模式，建立数据库及其应用系统，使之能够有效地存储数据，满足各种用户的应用需求。

数据库设计是一项涉及多学科的综合技术，同时，又是一项庞大的工程项目。数据库设计要和应用系统设计相结合，整个设计过程中要把结构设计和行为结合起来。早期的数据库设计完全取决于设计者的经验，设计过程是非结构化的，不能满足应用的要求。实践表明，数据库应用系统的开发是一项软件工程，开发过程应遵循软件工程的一般原则与方法，但又有其独特的一面。传统的软件工程忽视对应用中数据语义的分析和抽象。如，结构化设计和逐步求精的方法着重于处理过程的特性，只要有可能就尽量推迟数据结构设计的决策。这种方法对于数据库应用系统来说是不合适的。

由于信息结构复杂，应用环境多样，因而在相当长的一段时期内数据库设计主要采用手工试凑法。采用这种方法，数据库的质量与设计人员的经验和水平有直接关系，难以保证设计质量。十多年来，经过众多软件专业人员的努力探索，逐步形成了各种数据库设计方法，这些方法运用软件工程的思想和方法，提出了各种设计准则和规程。

这些规范的设计方法中比较著名的有新奥尔良方法、基于 E-R 模型的数据设计方法、基于 3NF 的设计方法、基于抽象语法规则的设计方法等等。这些是在数据库设计的不同阶段上支持实现的具体技术和方法。

数据库应用系统的中心问题是数据库结构的设计，按规范设计的方法，数据库设计分为以下几个阶段。

1. 需求分析阶段

设计一个有效的数据库必须要用系统的观点考虑问题。这一阶段系统分析员和用户双方共同收集数据库所需的信息内容和用户对处理的要求。在需求分析调研中，首先必须了解正在设计中的数据库所管理的数据将覆盖哪些部门，每个部门的数据来自何处，它们是依照何种原则处理数据，处理完毕后将输出信息到其它部门。然后，需和用户经过充分的讨论，确定数据库所要进行的数据处理的范围，确定哪些工作由人工完成，确定人机接口的界面。

这一阶段的工作是否能准确反映实际系统的信息流程情况和用户对系统的要求，影响到以后各阶段的工作，以及数据库系统将来运行的效率，因此，需求分析阶段的工作是整个数据库设计的基础。

2. 概念结构设计阶段

概念结构所涉及到的数据是独立于硬件和软件系统的，它的目标是以用户可以理解的形式来表达信息的流程，从而可以和不熟悉计算机的用户交换意见。这个阶段所得的概念结构要能充分地反映实体间的联系，成为反映现实的概念数据模型。这是各种基本数据模型的共同基础，易于向关系模型转换。

此阶段的设计工作分为两步，首先根据分析阶段所收集到的用户在数据和处理上的需求，得到每个用户各自的局部视图，对每个用户的局部数据结构进行描述。然后，利用软件工具分析各个局部视图，并把它们合并成统一的全局数据结构，即全局视图。

3. 逻辑结构设计阶段

此阶段的任务是把概念结构转换成数据库管理系统能处理的数据模型。在进行这种转换时，是把实体类型和联系类型分别转换成一个个关系模式，这中间存在着多种的可能组合，必须从中选取一个性能好的关系模式集作业关系数据库的模式。

这一阶段同时进行应用程序的设计，但目标只是开发应用程序的梗概。

4. 数据库的物理结构设计阶段

物理结构设计指对于给定的基本数据模型选择一个最适合应用环境的物理结构的过程。数据库的物理结构主要指数据库的存储记录格式、存储记录安排和存取方法，这些都依