

21 世纪高等院校教材

Visual FoxPro 程序设计教程

梁锐城 主编



科学出版社

www.sciencep.com

21 世纪高等院校教材

Visual FoxPro 程序设计教程

梁锐城 主编

科学出版社

北京

内 容 简 介

Visual FoxPro 6.0 是以可视化方式开发数据库应用程序的有力工具,它简单易学、功能强大。本教材从初学者的角度出发,结合编者多年的教学实践和编程经验,由浅入深、循序渐进地介绍了数据库的基础知识、数据库的基本操作、查询与视图、结构化查询语言 SQL、结构化程序设计、Visual FoxPro 6.0 的面向对象程序设计、菜单设计和报表设计等内容。

本书是依据新的《全国计算机等级考试二级考试大纲(Visual FoxPro 程序设计)》编写的,每章均提供相应的习题与上机操作题,可操作性强,适合作为高等院校本、专科学生的教材,也可以作为数据库应用系统开发人员和自学者的参考书籍。

图书在版编目(CIP)数据

Visual FoxPro 程序设计教程/梁锐城编著. —北京:科学出版社, 2004
21 世纪高等院校教材

ISBN 7-03-013248-3

I. V... II. 梁... III. 关系数据库—数据库管理系统, Visual FoxPro
6.0—高等学校—教材 IV. TP311.138

中国版本图书馆 CIP 数据核字(2004)第 028853 号

策划编辑: 孟战龙/责任编辑: 赵卫江

责任印制: 吕春珉/封面设计: 王 浩

科学出版社 出版

北京东黄城根北街16号

邮政编码: 100717

<http://www.sciencep.com>

双青印刷厂 印刷

科学出版社总发行 各地新华书店经销

*

2004 年 4 月第 一 版 开本: 787×1092 1/16

2004 年 4 月第一次印刷 印张: 19

印数: 1—4 000 字数: 437 000

定价: 26.00 元

(如有印装质量问题,我社负责调换(环伟))

前 言

随着计算机的日益普及,数据库技术已被广泛地应用于各个领域,学习和掌握数据库技术,已成为广大计算机使用者的普遍要求。

Visual FoxPro 6.0 是在微型计算机上最流行的新一代小型数据库管理系统之一,它以强大的功能,完善、丰富的工具,可靠、高效的管理方式,友好的界面,简单易学,便于开发为主要特点,深受许多小型数据库应用系统开发者的喜爱。

本书较系统地介绍了 Visual FoxPro 6.0 的基本知识,并以简明的例子讲解数据库管理、SQL 查询语言和程序设计,用丰富的实例演示各种数据库对象的具体操作和数据库应用系统的设计。本书的实例和习题皆以“学生管理”数据库为操作实体,易为读者理解和接受。

全书分 10 章,体现了新的《全国计算机等级考试二级考试大纲(Visual FoxPro 程序设计)》的要求,既不超大纲,也不降低要求。本书是高等院校的教学用书和参加全国计算机等级考试二级考试(Visual FoxPro)考生的参考书,也可供各类计算机培训班和个人自学使用。

本书的特点是内容新,实例丰富,由浅入深,最后形成一个完整的应用系统。本书所用例题和习题,是作者在多年教学实践中精选出来的,经过数届学生的使用、实践,对提高学生计算机应用能力有明显的效果。

本书第 1 章由沈少龙、梁锐城和魏森新编写,第 2 章到第 7 章、第 9 章由梁锐城编写,第 8 章、第 10 章由梁锐城、杨格和谢国海编写。最后由梁锐城统编全书。

为了便于读者学习,本书还免费提供电子教案 CAI 课件。读者可通过以下方法与我们联系。电话:0754-2902378 / 2902950, Email: rcliang@stu.edu.cn。

由于作者水平有限,书中难免存在不当之处,恳请读者批评指正。

作 者

2003 年 11 月

目 录

第1章 Visual FoxPro 基础	1
1.1 数据库基础知识	1
1.1.1 信息、数据与数据处理	1
1.1.2 数据模型	4
1.1.3 数据库应用系统	7
1.1.4 数据库管理系统	8
1.1.5 数据库系统的体系结构	13
1.2 Visual FoxPro 概述	14
1.2.1 Visual FoxPro 的特性	14
1.2.2 Visual FoxPro 的安装、启动与退出	16
1.2.3 Visual FoxPro 系统环境的配置	17
1.2.4 Visual FoxPro 用户界面	19
1.2.5 Visual FoxPro 的工作方式	21
1.2.6 项目管理器	21
1.2.7 Visual FoxPro 的向导、设计器、生成器	26
习题 1	29
第2章 自由表的基本操作	32
2.1 Visual FoxPro 的语言基础	32
2.1.1 数据类型	32
2.1.2 数据存储	33
2.1.3 函数	39
2.1.4 表达式	47
2.2 自由表的建立与修改	52
2.2.1 表结构的建立	52
2.2.2 表数据的输入	56
2.2.3 备注字段和通用型字段数据的输入和修改	58
2.2.4 表结构的修改	59
2.3 自由表的维护	59
2.3.1 Visual FoxPro 命令结构及常用子句	59
2.3.2 表的打开与关闭	61
2.3.3 表数据的浏览和修改	62
2.3.4 记录指针的移动	65

2.3.5	表与表结构的复制	71
2.3.6	记录的插入与追加	74
2.3.7	记录的删除与恢复	75
2.3.8	表数据的替换	78
2.3.9	设置过滤器和字段表	80
2.3.10	表数据的统计	82
习题 2		84
第 3 章	创建和使用数据库	89
3.1	建立、修改和删除数据库	89
3.1.1	建立数据库	89
3.1.2	修改数据库	90
3.1.3	删除数据库	92
3.2	打开和关闭数据库	93
3.3	创建数据库表	95
3.3.1	添加自由表	95
3.3.2	从数据库中移出表	97
3.3.3	新建数据库表	97
3.3.4	数据库表的属性设置	98
3.4	数据表的索引	101
3.4.1	索引的基本概念	101
3.4.2	创建索引	102
3.4.3	索引文件的打开与关闭	106
3.4.4	确定主控索引	106
3.4.5	删除索引	108
3.5	指针定位查询	108
3.5.1	顺序查询	108
3.5.2	索引查询	109
3.6	工作区与数据工作期	110
3.6.1	工作区	110
3.6.2	数据工作期	112
3.7	关联	114
3.7.1	永久性关联	114
3.7.2	临时性关联	116
3.7.3	数据表之间的参照完整性设置	119
习题 3		121
第 4 章	查询与视图	123
4.1	查询	123
4.1.1	建立查询	123
4.1.2	修改查询	129

871	4.1.3 运行查询	129
871	4.2 视图	131
871	4.2.1 建立视图	131
871	4.2.2 视图与数据更新	133
871	4.2.3 使用视图	134
181	习题 4	135
	第 5 章 结构化查询语言 SQL	136
881	5.1 SQL 的特点	136
881	5.2 SQL 的查询功能	137
781	5.2.1 简单查询	138
091	5.2.2 简单的联接查询	139
191	5.2.3 嵌套查询	139
891	5.2.4 排序	140
791	5.2.5 简单的计算查询	141
791	5.2.6 分组与计算查询	141
005	5.2.7 内外层互相关嵌套查询	142
005	5.2.8 使用量词和谓词的查询	142
505	5.2.9 超联接查询	144
805	5.2.10 集合的并运算	144
805	5.2.11 SQL SELECT 的特殊选项	145
805	5.3 SQL 的定义功能	146
015	5.3.1 表的定义	146
015	5.3.2 表的删除	148
515	5.3.3 表结构的修改	148
815	5.3.4 视图的定义及删除	149
815	5.4 SQL 的操作功能	151
815	5.4.1 插入	151
025	5.4.2 更新	152
125	5.4.3 删除	152
825	习题 5	153
	第 6 章 Visual FoxPro 的结构化程序设计	156
035	6.1 程序与程序文件	156
735	6.1.1 程序的概念	156
935	6.1.2 程序文件的建立与执行	156
045	6.1.3 简单的输入输出命令	158
845	6.2 程序的基本结构	163
245	6.2.1 选择结构	163
245	6.2.2 循环结构	167
745	6.3 模块化程序设计	173

6.3.1	子程序与模块的调用	173
6.3.2	过程	174
6.3.3	参数传递	176
6.3.4	用户自定义函数	178
6.3.5	变量的作用域	178
习题 6		181
第 7 章	Visual FoxPro 面向对象的程序设计与表单设计	186
7.1	面向对象程序设计基础	186
7.1.1	对象	186
7.1.2	类	187
7.1.3	事件和方法程序	190
7.1.4	Visual FoxPro 类的创建	191
7.1.5	对象的建立和使用	193
7.2	表单的基本操作	197
7.2.1	创建表单	197
7.2.2	修改表单	200
7.2.3	运行表单	200
7.3	表单设计器	202
7.3.1	表单设计器环境	203
7.3.2	控件的操作与布局	206
7.3.3	数据环境	208
7.4	常用表单控件	210
7.4.1	标签 (Label) 控件	210
7.4.2	命令按钮 (CommandButton) 控件	212
7.4.3	命令组 (CommandGroup) 控件	213
7.4.4	文本框 (TextBox) 控件	215
7.4.5	编辑框 (EditBox) 控件	218
7.4.6	选项组 (OptionGroup) 控件	220
7.4.7	复选框 (CheckBox) 控件	221
7.4.8	列表框 (ListBox) 控件	223
7.4.9	组合框 (ComboBox) 控件	225
7.4.10	微调框控件	226
7.4.11	表格 (Grid) 控件	227
7.4.12	页框 (PageFrame) 控件	229
7.4.13	表单控件的其他按钮	230
习题 7		233
第 8 章	菜单设计与应用	235
8.1	Visual FoxPro 系统菜单	235
8.2	设计下拉式菜单	237

285	8.2.1 调用菜单设计器	237
195	8.2.2 定义菜单	238
295	8.2.3 生成菜单程序	243
	8.2.4 运行菜单程序	243
	8.3 设计快捷菜单	245
	习题 8	247
	第 9 章 报表设计	248
	9.1 报表的基本操作	248
	9.1.1 创建报表文件	248
	9.1.2 修改报表	251
	9.1.3 报表的预览和打印	252
	9.2 在报表设计器中设计报表	254
	9.2.1 报表工具栏	254
	9.2.2 报表的数据源和布局	255
	9.2.3 在报表中使用控件	257
	9.2.4 数据分组和多栏报表	261
	习题 9	262
	第 10 章 应用程序的开发	264
	10.1 应用程序开发的一般过程	264
	10.1.1 需求分析	264
	10.1.2 数据库设计	265
	10.1.3 应用程序设计	266
	10.1.4 软件测试	266
	10.1.5 应用程序发布	267
	10.1.6 系统运行和维护	267
	10.2 管理系统开发实例	267
	10.2.1 规划学生管理系统的功能结构	267
	10.2.2 数据库设计	268
	10.2.3 设计应用程序	268
	10.2.4 系统主菜单设计	269
	10.2.5 设计主程序	270
	10.2.6 连编应用程序	272
	10.2.7 运行应用程序	275
	10.3 应用程序生成器	276
	10.3.1 使用应用程序向导	276
	10.3.2 应用程序生成器及其使用	278
	习题 10	283
	附录	285

附录 1	属性名称与功能	285
附录 2	方法名称与功能	291
附录 3	事件名称与功能	292
第 8 章	菜单设计	303
8.1	菜单设计概述	303
8.2	菜单设计步骤	304
8.3	菜单设计实例	305
第 9 章	表单设计	310
9.1	表单设计概述	310
9.2	表单设计步骤	311
9.3	表单设计实例	312
第 10 章	报表设计	317
10.1	报表设计概述	317
10.2	报表设计步骤	318
10.3	报表设计实例	319
第 11 章	数据库设计	324
11.1	数据库设计概述	324
11.2	数据库设计步骤	325
11.3	数据库设计实例	326
第 12 章	网络应用开发	331
12.1	网络应用概述	331
12.2	网络应用设计	332
12.3	网络应用实例	333
第 13 章	应用系统开发	338
13.1	应用系统概述	338
13.2	应用系统设计	339
13.3	应用系统开发	340
13.4	应用系统维护	341
13.5	应用系统测试	342
13.6	应用系统部署	343
13.7	应用系统升级	344
13.8	应用系统备份	345
13.9	应用系统恢复	346
13.10	应用系统安全	347
13.11	应用系统性能	348
13.12	应用系统兼容性	349
13.13	应用系统兼容性	350
13.14	应用系统兼容性	351
13.15	应用系统兼容性	352
13.16	应用系统兼容性	353
13.17	应用系统兼容性	354
13.18	应用系统兼容性	355
13.19	应用系统兼容性	356
13.20	应用系统兼容性	357
13.21	应用系统兼容性	358
13.22	应用系统兼容性	359
13.23	应用系统兼容性	360
13.24	应用系统兼容性	361
13.25	应用系统兼容性	362
13.26	应用系统兼容性	363
13.27	应用系统兼容性	364
13.28	应用系统兼容性	365
13.29	应用系统兼容性	366
13.30	应用系统兼容性	367
13.31	应用系统兼容性	368
13.32	应用系统兼容性	369
13.33	应用系统兼容性	370
13.34	应用系统兼容性	371
13.35	应用系统兼容性	372
13.36	应用系统兼容性	373
13.37	应用系统兼容性	374
13.38	应用系统兼容性	375
13.39	应用系统兼容性	376
13.40	应用系统兼容性	377
13.41	应用系统兼容性	378
13.42	应用系统兼容性	379
13.43	应用系统兼容性	380
13.44	应用系统兼容性	381
13.45	应用系统兼容性	382
13.46	应用系统兼容性	383
13.47	应用系统兼容性	384
13.48	应用系统兼容性	385
13.49	应用系统兼容性	386
13.50	应用系统兼容性	387
13.51	应用系统兼容性	388
13.52	应用系统兼容性	389
13.53	应用系统兼容性	390
13.54	应用系统兼容性	391
13.55	应用系统兼容性	392
13.56	应用系统兼容性	393
13.57	应用系统兼容性	394
13.58	应用系统兼容性	395
13.59	应用系统兼容性	396
13.60	应用系统兼容性	397
13.61	应用系统兼容性	398
13.62	应用系统兼容性	399
13.63	应用系统兼容性	400
13.64	应用系统兼容性	401
13.65	应用系统兼容性	402
13.66	应用系统兼容性	403
13.67	应用系统兼容性	404
13.68	应用系统兼容性	405
13.69	应用系统兼容性	406
13.70	应用系统兼容性	407
13.71	应用系统兼容性	408
13.72	应用系统兼容性	409
13.73	应用系统兼容性	410
13.74	应用系统兼容性	411
13.75	应用系统兼容性	412
13.76	应用系统兼容性	413
13.77	应用系统兼容性	414
13.78	应用系统兼容性	415
13.79	应用系统兼容性	416
13.80	应用系统兼容性	417
13.81	应用系统兼容性	418
13.82	应用系统兼容性	419
13.83	应用系统兼容性	420
13.84	应用系统兼容性	421
13.85	应用系统兼容性	422
13.86	应用系统兼容性	423
13.87	应用系统兼容性	424
13.88	应用系统兼容性	425
13.89	应用系统兼容性	426
13.90	应用系统兼容性	427
13.91	应用系统兼容性	428
13.92	应用系统兼容性	429
13.93	应用系统兼容性	430
13.94	应用系统兼容性	431
13.95	应用系统兼容性	432
13.96	应用系统兼容性	433
13.97	应用系统兼容性	434
13.98	应用系统兼容性	435
13.99	应用系统兼容性	436
14.00	应用系统兼容性	437

第1章 Visual FoxPro 基础

Visual FoxPro 是目前微型计算机上优秀的数据库管理系统之一，正如其名称中冠之的 Visual 一样，它采用了可视化的、面向对象的程序设计方法，大大简化了应用系统的开发过程。微机数据库系统以其开发成本低、简单易学、方便用户等优点得到迅速推广。计算机应用人员只有掌握数据库系统的基础知识，熟悉数据库管理系统的特点，才能开发出适用的数据库应用系统。本章将介绍数据库的基本概念和关系数据库的基础知识，掌握这些内容是学好、用好 Visual FoxPro 的必要前提条件。

1.1 数据库基础知识

数据库是计算机系统的一个重要组成部分。数据库技术是信息技术中应用最广泛的技术之一，如图 1.1 所示。

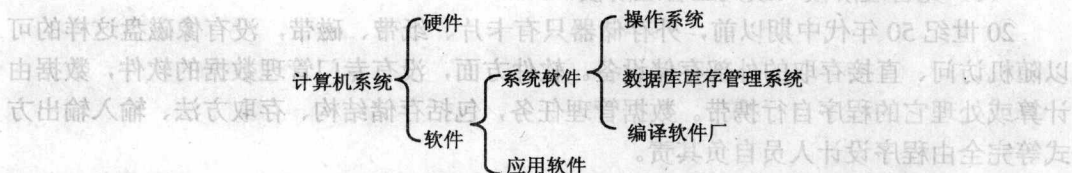


图 1.1 数据库的作用

1.1.1 信息、数据与数据处理

数据库管理系统是处理数据的有效工具，因此首先需要了解数据、数据处理的概念和计算机数据管理的发展历程。

1. 数据与数据处理

数据是指存储在某一媒体上能够识别的物理符号。数据的概念包括两个方面：其一是描述事物特性的数据内容；其二是存储在某一媒体上的数据形式。数据形式可以是多种多样的，例如，某人的出生日期是“1980 年 9 月 5 日”，也可以表示为“80 / 09 / 05”，其含义并没有改变。

数据的概念在数据处理领域中已经大大地拓宽了。数据不仅包括数字、字母、文字和其他特殊字符组成的文本形式的数据，而且还包括图形、图像、动画、影像、声音等

多媒体数据。但是使用最多、最基本的仍然是文字数据。

数据处理是对各种类型的数据进行收集、存储、分类、计算、加工、检索和传输的过程。可以说是将数据转换成信息的过程。从数据处理的角度而言,信息是一种被加工成特定形式的数据,这种数据形式对于数据接收者来说是有意义的。

人们有时说“信息处理”,其真正含义应该是为了产生信息而处理数据。通过处理数据可以获得信息,通过分析和筛选信息可以产生决策。例如,一个人的“出生日期”是有生以来不可改变的基本特征之一,属于原始数据,而“年龄”是通过现年与出生日期相减的简单计算而得到的二次数据。根据年龄、性别、职称等有关一个人的信息和离退休年龄的规定,可以判断此人何时应当办理离退休手续。

在计算机中,使用计算机外存储器,如磁盘来存储数据;通过计算机软件来管理数据;通过应用程序来对数据进行加工处理。

2. 计算机数据管理

数据处理的中心问题是数据管理。计算机对数据的管理是指对数据的组织、分类、编码、存储、检索和维护提供操作手段。

计算机在数据管理方面也经历了由低级到高级的发展过程。计算机数据管理随着计算机硬件、软件技术和计算机应用范围的发展而不断发展,多年来经历了人工管理、文件系统、数据库系统、分布式数据库系统和面向对象数据库系统等几个阶段。

(1) 无管理阶段(或人工管理阶段)

20 世纪 50 年代中期以前,外存储器只有卡片、纸带、磁带,没有像磁盘这样的可以随机访问、直接存取的外部存储设备。软件方面,没有专门管理数据的软件,数据由计算或处理它的程序自行携带。数据管理任务,包括存储结构、存取方法、输入输出方式等完全由程序设计人员自负其责。

这一时期计算机数据管理的特点是:数据与程序不具有独立性,一组数据对应一组程序。数据不长期保存,程序运行结束后就退出计算机系统,一个程序中的数据无法被其他程序利用。因此程序与程序之间存在大量的重复数据,称为数据冗余。

(2) 文件系统

20 世纪 50 年代后期至 60 年代中后期,计算机开始大量地用于管理中的数据处理工作,大量的数据存储、检索和维护成为紧迫的需求。可直接存取的磁盘成为联机的主要外存。在软件方面,出现了高级语言和操作系统。操作系统中的文件系统是专门管理外存储器的数据管理软件。

在文件系统阶段,程序与数据有了一定的独立性,程序和数据分开存储,有了程序文件和数据文件的区别。数据文件可以长期保存在外存储器上被多次存取。

在文件系统的支持下,程序只需用文件名访问数据文件,程序员可以集中精力在数据处理的算法上,而不必关心记录在存储器上的地址和内、外存交换数据的过程。

但是,文件系统中的数据文件是为了满足特定业务领域或某部门的专门需要而设计的,服务于某一特定应用程序,数据和程序相互依赖。同一数据项可能重复出现在多个文件中,导致数据冗余度大,这不仅浪费存储空间,增加更新开销,更严重的是,由于不能统一修改,容易造成数据的不一致。

文件系统存在的问题阻碍了数据处理技术的发展,不能满足日益增长的信息需求,这正是数据库技术产生的原动力,也是数据库系统产生的背景。

(3) 数据库系统

从 20 世纪 60 年代后期开始,需要计算机管理的数据量急剧增长,并且对数据共享的需求日益增强。文件系统的数据管理方法已无法适应开发应用系统的需要。为了实现计算机对数据的统一管理,达到数据共享的目的,数据库技术应运而生。

数据库技术的主要目的是有效地管理和存取大量的数据资源,包括:提高数据的共享性,使多个用户能够同时访问数据库中的数据;减小数据的冗余度,以提高数据的一致性和完整性;提供数据与应用程序的独立性,从而减少应用程序的开发和维护代价。

(4) 分布式数据库系统

分布式数据库系统是数据库技术和计算机网络技术紧密结合的产物。在 20 世纪 70 年代后期之前,数据库系统多数是集中式的。网络技术的发展为数据库提供了分布式运行环境。

数据库技术与网络技术的结合分为紧密结合与松散结合两大类。因此,分布式数据库管理系统分为物理上分布、逻辑上集中的分布式数据库结构和物理上分布、逻辑上分布的分布式数据库结构两种。

物理上分布、逻辑上集中的分布式数据库结构是一个逻辑上统一、地域上分布的数据集合,是计算机网络环境中各个结点局部数据库的逻辑集合,同时受分布式数据库管理系统的统一控制和管理。即把全局数据模式按数据来源和用途,合理分布在系统的多个结点上,使大部分数据可以就地或就近存取,而用户不感到数据的分布。

物理上分布、逻辑上分布的分布式数据库结构是把多个集中式数据库系统通过网络连接起来,各个结点上的计算机可以利用网络通信功能访问其他结点上的数据库资源。它一般由两部分组成:一是本地结点的数据,二是本地结点共享的其他结点上的有关的数据。在这种运行环境中,各个数据库系统的数据库由各自独立的数据库管理系统集中管理。结点间的数据共享由双边协商确定。这种数据库结构有利于数据库的集成、扩展和重新配置。

(5) 面向对象数据库系统

面向对象方法是一种认识、描述事物的方法论,它起源于程序设计语言。面向对象程序设计是 20 世纪 80 年代引入计算机科学领域的一种新的程序设计技术和范型,它的发展十分迅猛,影响涉及计算机科学及其应用的各个领域。

通俗地讲,面向对象的方法就是按照人们认识世界和改造世界的习惯方法对现实世界的客观事物/对象进行最自然的、最有效的抽象和表达,同时又以各种严格高效的行为规范和机制实施客观事物的有效模拟和处理,而且把对客观事物的表达(对象属性结构)和对它的操作处理(对象行为特征)结合成为一个有机整体,事物完整的内部结构和外部行为机制被反映得淋漓尽致。

面向对象数据库是数据库技术与面向对象程序设计相结合的产物。面向对象数据库是面向对象方法在数据库领域中的实现和应用,它既是一个面向对象的系统,又是一个数据库系统。Visual FoxPro 不但仍然支持标准的过程化程序设计,而且在语言上还进行了扩展,提供了面向对象程序设计的强大功能和更大的灵活性。本教程将在第 7 章中详

细介绍面向对象的基本概念。

1.1.2 数据模型

数据库需要根据应用系统中数据的性质和内在联系,按照管理的要求来设计和组织。人们把客观存在的事物以数据的形式存储到计算机中,经历了对现实生活中事物特性的认识、概念化到计算机数据库里的具体表示的逐级抽象过程。

1. 数据库

数据库(database)是以一定的组织形式存放在计算机存储介质上的相互关联的数据的集合。它不仅包括描述事物的数据本身,而且还包括相关事物之间的联系。

数据库的特点有:数据具有最小的冗余度,资源共享性,数据独立性,数据安全可靠,保密性能好。

数据库中的数据往往不是像文件系统那样,只面向某一项特定应用,而是面向多种应用,可以被多个用户、多个应用程序共享。数据结构独立于使用数据的程序,对于数据的增加、删除、修改和检索由系统软件进行统一的控制。

2. 实体的描述

现实世界存在各种事物,事物与事物之间存在着联系。这种联系是客观存在的,是由事物本身的性质所决定的。例如,图书馆中有图书和读者,读者借阅图书;学校的教学系统中有教师、学生、课程,教师为学生授课,学生选修课程并取得成绩;在物资或商业部门有货物、客户,客户要订货、购物。如果管理的对象较多或者比较特殊,事物之间的联系就可能较为复杂。

(1) 实体

客观存在并且可以相互区别的事物称为实体。实体可以是实际的事物,也可以是抽象的事件。比如,职工、图书等属于实际事物,订货、借阅图书、比赛等活动是比较抽象的事件。

(2) 实体的属性

描述实体的特性称为属性。例如,职工实体用(职工号,姓名,性别,出生日期,职称)等若干个属性来描述;图书实体用(总编号,分类号,书名,作者,单价)等属性来描述。

(3) 实体集和实体型

字段值的集合表示一个实体,而属性的集合表示一种实体的类型,称为实体型。同类型的实体的集合,称为实体集。

例如,在职工实体集当中,(1069,张大道,男,65/12/06,教授)表征教工名册中的一个具体人;在图书实体集中,(10765,TP196,Visual FoxPro 教程,吴桂,22.50)则具体代表一本书。

在 Visual FoxPro 中,用“表”来存放同一类实体,即实体集。例如,职工表、图书

表等。Visual FoxPro 的一个表包含若干个字段,表中所包含的“字段”就是实体的属性。字段值的集合组成表中的一条记录,代表一个具体的实体,即每一条记录表示一个实体。

3. 实体间联系及联系的种类

实体之间的对应关系称为联系,它反映现实世界事物之间的相互关联。例如,一位读者可以借阅若干本图书;同一本书可以相继被几个读者借阅。

实体间联系的种类是指一个实体型中可能出现的每一个实体与另一个实体型中多个具体实体存在联系。两个实体间的联系可以归结为三种类型:

(1) 一对一联系 (one-to-one relationship)

考查公司和总经理两个实体型,如果一个公司只有一个总经理,一个总经理不能同时在其他公司再兼任总经理,在这种情况下公司和总经理之间存在一对一的联系。

在 Visual FoxPro 中,一对一的联系表现为主表中的每一条记录只与相关表中的一条记录相关联。例如,一个单位劳资部门的职工表和财务部门使用的工资表之间就存在一对一的联系。

(2) 一对多联系 (one-to-many relationship)

考查部门和职工两个实体型,一个部门有多名职工,而一名职工只在一个部门就职,即只占一个部门的编制。部门与职工之间则存在一对多的联系。考查学生和系两个实体集,一个学生只能在一个系里注册,而一个系有很多个学生。系和学生也是一对多的联系。

在 Visual FoxPro 中,一对多的联系表现为主表中的每一条记录与相关表中的多条记录相关联。即表 A 的一个记录在表 B 中可以有多个记录与之对应,但表 B 中的一个记录最多只能有一个表 A 的记录与之对应。

一对多联系是最普遍的联系。也可以把一对一的联系看成一对多联系的一个特殊情况。

(3) 多对多联系 (many-to-many relationship)

考查学生和课程两个实体型,一个学生可以选修多门课程,一门课程由多个学生选修。因此,学生和课程间存在多对多的联系。图书与读者之间也是多对多联系,因为一位读者可以借阅若干本图书,同一本书可以相继被几个读者借阅。

在 Visual FoxPro 中,多对多的联系表现为一个表中的多个记录在相关表中同样有多个记录与其匹配。即表 A 的一条记录在表 B 中可以对应多条记录,而表 B 的一条记录在表 A 中也可以对应多条记录。

4. 数据模型简介

为了反映事物本身及事物之间的各种联系,数据库中的数据必须有一定的结构,这种结构用数据模型来表示。数据库不仅管理数据本身,而且要使用数据模型表示出数据之间的联系。可见,数据模型是数据库管理系统用来表示实体及实体间联系的方法。一个具体的数据模型应当正确地反映出数据之间存在的整体逻辑关系。

任何一个数据库管理系统都是基于某种数据模型的。数据库管理系统所支持的数据模型分为三种:层次模型、网状模型、关系模型。因此,使用支持某种特定数据模型的

数据库管理系统开发出来的应用系统相应地称为层次数据库系统、网状数据库系统、关系数据库系统。

关系模型对数据库的理论和实践产生很大的影响，成为当今最流行的数据库模型。本书重点介绍关系数据库的基本概念和使用，为了使读者对数据模型有一个全面的认识，进而更深刻地理解关系模型，这里先对层次模型和网状模型做一些简单的介绍，然后再比较详细地介绍关系数据模型。

(1) 层次数据模型

用树形结构表示实体及其之间联系的模型称为层次模型。在这种模型中，数据被组织成由“根”开始的“树”，每个实体由根开始沿着不同的分支放在不同的层次上。如果不再向下分支，那么此分支序列中最后的结点称为“叶”。上级结点与下级结点之间为一对多的联系，图 1.2 给出一个层次模型的例子。

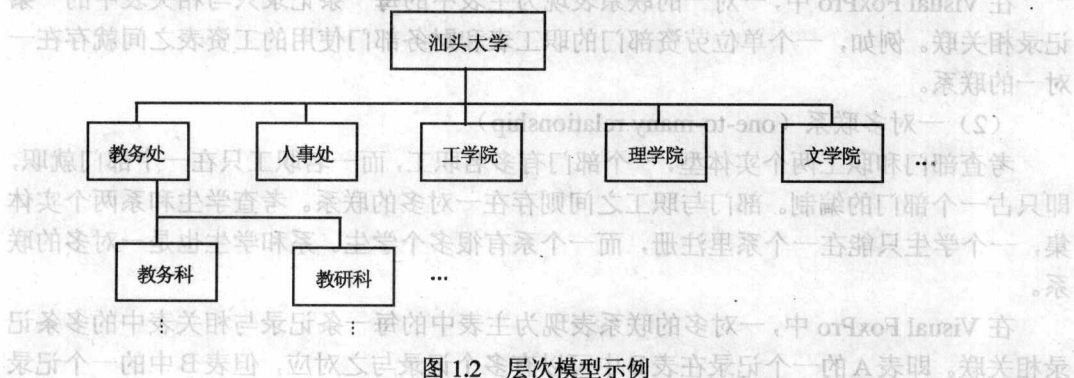


图 1.2 层次模型示例

层次模型实际上是由若干个代表实体之间一对多联系的基本层次联系组成的一棵树，树的每一个结点代表一个实体类型。从图 1.2 中可以看出，“汕头大学”是根结点，“汕头大学”管理的树状结构反映的是实体型之间的结构。该模型的实际存储数据由链接指针来体现联系。

支持层次数据模型的数据库管理系统称为层次数据库管理系统，在这种系统中建立的数据库是层次数据库。层次数据模型不能直接表示出多对多的联系。

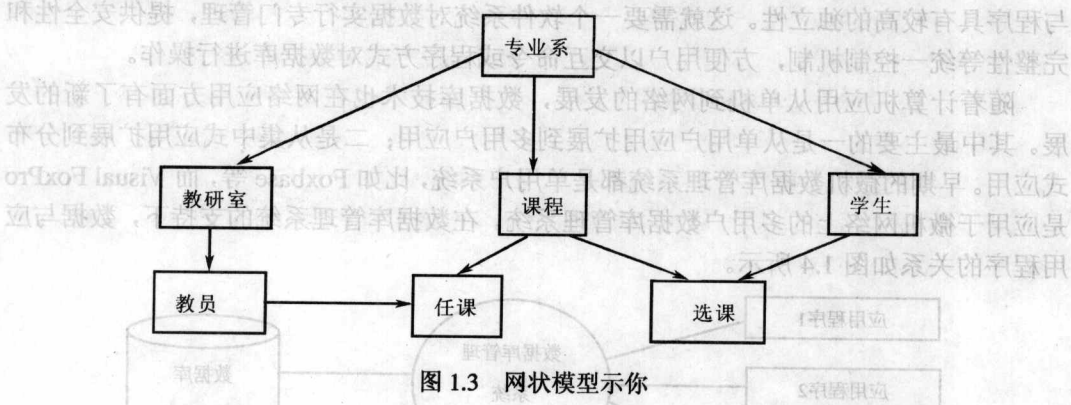
(2) 网状模型

用网状结构表示实体及其之间联系的模型称为网状模型。网中的每一个结点代表一个实体类型。网状模型突破了层次模型的两点限制：允许结点有多于一个的父结点；可以有一个以上的结点没有父结点。因此，网状模型可以方便地表示各种类型的联系。

图 1.3 给出了一个简单的网状模型。每一个联系都代表实体之间一对多的联系，系统用单向或双向环形链接指针来具体实现这种联系。如果课程和选课人数较多，链接将变得相当复杂。网状模型的主要优点是表示多对多的联系具有很大的灵活性，这种灵活性是以数据结构复杂化为代价的。

支持网状数据模型的数据库管理系统称为网状数据库管理系统，在这种系统中建立的数据库是网状数据库。网状模型和层次模型在本质上是一样的。从逻辑上看，它们都是用结点表示实体，用有向边（箭头）表示实体间的联系，实体和联系用不同的方法来

表示;从物理上看,每一个结点都是一个存储记录,用链接指针来实现记录之间的联系。这种用指针将所有数据记录都“捆绑”在一起的特点,使得层次模型和网状模型存在难以实现系统的修改与扩充等缺陷。



(3) 关系数据模型

用二维表结构来表示实体以及实体之间联系的模型称为关系模型。关系数据模型是以关系数学理论为基础的。在关系模型中,操作的对象和结果都是二维表,这种二维表就是关系。

关系模型与层次型、网状型的本质区别在于数据描述的一致性,模型概念单一。在关系型数据库中,每一个关系都是一个二维表,无论实体本身还是实体间的联系均用称为“关系”的二维表来表示,使得描述实体的数据本身能够自然地反映它们之间的联系。而传统的层次和网状模型数据库是使用链接指针来存储和体现联系的。

关系数据模型的特点:

① 一个二维表中,所有的记录格式相同、长度相同。

② 同一字段数据的性质是相同的,它们均为同一属性的值。

③ 行和列的排列顺序并不重要。

关系数据库以其完备的理论基础、简单的模型、说明性的查询语言和使用方便等优点得到广泛应用。

1.1.3 数据库应用系统

数据库应用系统是指系统开发人员利用数据库系统资源开发出来的,面向某一类实际应用的应用软件系统。随着社会上各行业急剧增长的数据处理的需要,为提高数据处理速度、工作效率和经济效益,人们利用数据库技术开发了大量的应用程序,如:POS 电脑收银系统、人事管理系统、图书管理系统、教学管理系统、生产管理系统,以及工资、财务管理系统等。无论是面向内部业务和管理的管理信息系统,还是面向外部,提供信息服务的开放式信息系统,从实现技术角度而言,都是以数据库为基础和核心的计算机应用系统。

一个数据库应用系统通常由计算机软硬件系统、数据库管理系统、应用程序和反映