



Visual FoxPro 6.0 程序设计

(修订版)

主 编 齐智敏 许 杰
副主编 吴学芝 董晓莹



NEUPRESS
东北大学出版社

21 世纪新理念高职高专规划教材



Visual FoxPro 6.0 程序设计

(修 订 版)

主 编 齐智敏 许 杰

副主编 吴学芝 董晓莹

东北大学出版社

• 沈 阳 •

© 东北大学出版社 2004

图书在版编目 (CIP) 数据

Visual FoxPro 6.0 程序设计 / 齐智敏, 许杰主编. —2 版. —沈阳: 东北大学出版社, 2008.1

(21 世纪新理念高职高专规划教材)

ISBN 978-7-81054-998-1

I.V… II.①齐… ②许… III.关系数据库—数据库管理系统, Visual FoxPro 6.0—程序设计 IV.TP311.138

中国版本图书馆 CIP 数据核字 (2008) 第 009805 号

出 版 者: 东北大学出版社

地址: 沈阳市和平区文化路 3 号巷 11 号

邮编: 110004

电话: 024—83687331 (市场部) 83680267 (社务室)

传真: 024—83680180 (市场部) 83680265 (社务室)

E-mail: neuph @ neupress.com

http: // www. neupress. com

印 刷 者: 沈阳中科印刷有限责任公司

发 行 者: 东北大学出版社

幅面尺寸: 184mm × 260mm

印 张: 20

字 数: 512 千字

出版时间: 2004 年 2 月第 1 版

2008 年 1 月第 2 版

印刷时间: 2008 年 1 月第 2 次印刷

责任编辑: 王兆元

封面设计: 唐敏智

责任校对: 章 力

责任出版: 杨华宁

ISBN 978-7-81054-998-1

定 价: 29.00 元

前 言

随着计算机技术的发展和普及, 各行各业的管理部门需要用计算机处理大量的信息。选择一个优秀的数据库管理系统作为开发平台, 将给日后的信息处理带来极大方便。Microsoft Visual FoxPro 6.0 关系数据库系统作为新一代小型数据库管理系统的杰出代表, 它以强大的性能、完整而又丰富的工具、较高的处理速度、友好的界面以及完备的兼容性等特点, 备受广大用户的欢迎。Visual FoxPro 6.0 提供了一个集成化的系统开发环境, 它不仅支持过程式编程技术, 而且在语言方面作了强大的扩充, 支持面向对象的可视化编程技术, 支持网络环境的应用, 并拥有功能强大的可视化程序设计工具。利用 Visual FoxPro 6.0 可以设计出丰富多彩的用户界面, 在用户界面中可以放置各种控制部件, 如命令按钮、图形图片、图表等, 从而设计出完全图形化的界面, 方便用户的操作和使用。

Visual FoxPro 6.0 是可以运行于 Windows 9x 和 Windows NT 平台的 32 位数据库开发系统, Visual FoxPro 6.0 使组织数据、定义数据库规则和建立应用程序等工作变得简单易行。利用可视化的设计工具和向导, 用户可以快速创建表单、查询和打印报表。目前 Visual FoxPro 6.0 已成为用户收集信息、查询数据、创建数据、创建集成数据库系统、进行应用系统开发较为理想的工具软件。

本书针对初学者的特点, 在体系结构和内容上注意了由简入繁、由浅入深、循序渐进、深入浅出以及理论与实践相结合。全书包括正文 14 章及附录。第 1 至 6 章介绍了 Visual FoxPro 6.0 编程环境、编程语言、编程方法等编程基本知识; 第 7 至 13 章介绍了面向对象的编程技术; 第 14 章为综合开发实例; 附录为计算机等级考试模拟测试题。每一章均提供相应的上机指导和习题, 具有很强的针对性和实用性。

本书第 1, 2, 12, 13 章由齐智敏(辽东学院)编写; 第 3, 4, 5 章由许杰(辽东学院)编写; 第 6, 11 章由吴学芝(辽宁信息职业技术学院)编写; 第 7, 8, 9, 10 章及附录由董晓莹(辽东学院)编写。

尽管在编写此书过程中作者做了许多努力, 但由于水平有限, 加之编写时间仓促, 书中缺点和疏漏之处在所难免, 请读者批评指正。

作 者

2007 年 12 月

目 录

第1章 Visual FoxPro 基础 1

1.1 数据库基础知识 1

1.1.1 数据库的发展 1

1.1.2 数据库系统 2

1.1.3 数据模型 3

1.1.4 关系数据库 5

1.2 Visual FoxPro 6.0 系统概述 7

1.2.1 Visual FoxPro 6.0 的特点 7

1.2.2 Visual FoxPro 6.0 的安装与启动 8

1.2.3 Visual FoxPro 6.0 的用户界面 9

1.2.4 Visual FoxPro 6.0 的环境配置 11

1.2.5 Visual FoxPro 6.0 的工作方式 13

1.2.6 Visual FoxPro 6.0 中文版的性能指标 14

1.2.7 Visual FoxPro 6.0 中的文件类型 14

1.3 项目管理器 15

1.3.1 创建项目 15

1.3.2 项目管理器的组成 16

1.3.3 项目管理器的使用 17

1.3.4 定制项目管理器 18

1.4 Visual FoxPro 的辅助设计工具 20

1.4.1 Visual FoxPro 的向导 20

1.4.2 Visual FoxPro 的设计器 20

1.4.3 Visual FoxPro 的生成器 20

习题 1 21

上机操作 1 22

第2章 数据与数据运算 23

2.1 数据类型 23

2.1.1 Visual FoxPro 6.0 的数据类型 23

2.1.2 字段数据类型 23

2.2 数据存储 24

2.3 常量与变量 24

2.3.1 常量 24

2.3.2 变量 27

2.3.3 数组 29

2.4 函 数 31

2.4.1 数值函数 32

2.4.2 字符函数 33

2.4.3 日期和时间函数 35

2.4.4 数据类型转换函数 36

2.4.5 测试函数 37

2.5 表达式 40

2.5.1 算术表达式 40

2.5.2 关系表达式 41

2.5.3 字符表达式 43

2.5.4 逻辑表达式 43

2.5.5 日期时间表达式 44

2.5.6 运算优先级 44

2.6 命令格式及书写规则 44

2.6.1 命令格式 44

2.6.2 书写规则 45

习题 2 45

上机操作 2 47

第3章 Visual FoxPro 数据库及其操作

..... 48

3.1 Visual FoxPro 数据库及其建立 48

3.1.1 基本概念 48

3.1.2 数据库的设计步骤 48

3.1.3 建立数据库 56

3.1.4 使用数据库 57

3.1.5 修改数据库 58

3.1.6 删除数据库 60

3.2 建立数据库表 61

3.2.1 在数据库中建立表 61

3.2.2 修改表结构 64

3.3 表的基本操作 65

3.3.1 使用浏览器操作表 65

3.3.2 增加记录的命令 67

3.3.3 删除记录的命令 67

3.3.4 修改记录的命令 68

3.3.5 显示记录的命令 69

3.3.6 查询定位命令 69

3.3.7 统计与计算数据 70

3.4 索 引 71

3.4.1 基本概念 72

3.4.2 在表设计器中建立索引 73

3.4.3 用命令建立索引 74

3.4.4 使用索引 77

3.5 数据完整性	78	4.4.4 视图的定义	114
3.5.1 实体完整性与主关键字	78	习题 4	116
3.5.2 域完整性与约束规则	78	上机操作 4	118
3.5.3 参照完整性与表之间的关联	79	第 5 章 查询与视图	119
3.6 自由表	82	5.1 查 询	119
3.6.1 数据库表与自由表	82	5.1.1 创建查询的步骤	119
3.6.2 将自由表添加到数据库	83	5.1.2 使用查询向导建立查询	119
3.6.3 从数据库中移出表	84	5.1.3 使用查询设计器创建查询	121
3.7 多个表的同时使用	85	5.1.4 查询设计器的局限性	124
3.7.1 多工作区的概念	85	5.1.5 运行查询	125
3.7.2 使用不同工作区的表	86	5.2 视 图	126
3.7.3 表之间的关联	86	5.2.1 创建本地视图	126
3.8 排 序	87	5.2.2 远程视图与连接	130
习题 3	87	5.2.3 视图与数据更新	131
上机操作 3	89	5.2.4 使用视图	132
第 4 章 关系数据库标准语言 SQL	93	习题 5	133
4.1 SQL 概述	93	上机操作 5	134
4.1.1 SQL 语言的主要特点	93	第 6 章 程序设计基础	135
4.1.2 SQL 语句的执行	94	6.1 程序与程序文件	135
4.2 查询功能	94	6.1.1 程序的概念	135
4.2.1 简单查询	95	6.1.2 程序文件的建立与运行	136
4.2.2 简单的联接查询	96	6.1.3 辅助命令及运行环境设置命令	137
4.2.3 嵌套查询	97	6.2 程序的基本结构	138
4.2.4 几个特殊的运算符	97	6.2.1 顺序结构程序设计	138
4.2.5 排 序	98	6.2.2 选择结构程序设计	139
4.2.6 简单的计算查询	99	6.2.3 循环结构	142
4.2.7 分组与计算查询	100	6.2.4 循环结构的嵌套	144
4.2.8 利用空值查询	101	6.2.5 循环结构中的 LOOP 和 EXIT 语句	145
4.2.9 别名与自联接查询	101	6.3 子程序、过程和自定义函数	146
4.2.10 内外层互相关嵌套查询	102	6.3.1 子程序	146
4.2.11 使用量词和谓词的查询	103	6.3.2 过程与过程文件	147
4.2.12 超联接查询	104	6.3.3 参数传递	150
4.2.13 集合的并运算	106	6.3.4 自定义函数	151
4.2.14 Visual FoxPro 中 SQL SELECT 的几个特殊选项	106	6.4 编程实例	152
4.3 操作功能	108	习题 6	154
4.3.1 插 入	108	上机操作 6	156
4.3.2 更 新	109	第 7 章 面向对象程序设计	157
4.3.3 删 除	110	7.1 对 象	157
4.4 定义功能	110	7.1.1 对象的属性	157
4.4.1 表的定义	110		
4.4.2 表的删除	112		
4.4.3 表结构的修改	113		

7.1.2 事件	159
7.1.3 方法程序	160
7.1.4 数据环境	161
7.1.5 对象的引用	162
7.2 类	162
7.2.1 类的概念	162
7.2.2 自定义类	164
7.2.3 创建工具栏	168
习题 7	171
上机操作 7	173

第 8 章 表单设计 174

8.1 创建表单	174
8.1.1 使用向导创建表单	174
8.1.2 使用表单设计器创建表单	179
8.1.3 使用“表单生成器”创建表单	179
8.1.4 保存和运行表单	180
8.2 创建表单集	181
8.2.1 应用程序界面	181
8.2.2 表单集	182
8.3 向表单中添加对象	183
8.3.1 添加 VFP 容器	183
8.3.2 添加 VFP 控件	183
8.3.3 添加自定义对象	184
8.4 定制表单	185
8.4.1 美化表单	185
8.4.2 表单的布局	186
8.4.3 设置 Tab 键次序	186
8.5 表单/表单集设计实例	187
8.5.1 一个统计示例	187
8.5.2 封面表单集示例	188
习题 8	190
上机操作 8	191

第 9 章 控件的使用 192

9.1 输出类控件	192
9.1.1 Label(标签)控件	192
9.1.2 Image(图像)控件	194
9.1.3 Line(线条)控件	196
9.1.4 Shape(形状)控件	197
9.2 输入类控件	199
9.2.1 TextBox(文本框)控件	199
9.2.2 EditBox(编辑框)控件	201
9.2.3 ListBox(列表框)控件	202

9.2.4 ComboBox(组合框)控件	205
9.2.5 Spinner(微调)控件	209
9.3 控制类控件	210
9.3.1 CommandButton(命令按钮)控件	210
9.3.2 CommandGroup(命令按钮组)控件	212
9.3.3 OptionGroup(选项按钮组)控件	212
9.3.4 CheckBox(复选框)控件	214
9.3.5 Timer(计时器)控件	216
9.4 容器类控件	217
9.4.1 Grid(表格)控件	217
9.4.2 PageFrame(页框)控件	219
9.4.3 Container(容器)控件	220
9.5 连接类控件	221
9.5.1 ActiveX 控件	222
9.5.2 ActiveX 绑定控件	224
9.5.3 超级链接	224
习题 9	224
上机操作 9	225

第 10 章 菜单设计 227

10.1 Visual FoxPro 的系统菜单	227
10.1.1 菜单的结构	227
10.1.2 系统菜单	227
10.2 下拉式菜单设计	228
10.2.1 创建下拉式菜单的步骤	228
10.2.2 快速创建菜单	230
10.2.3 菜单设计器窗口	230
10.2.4 显示菜单的命令	232
10.2.5 创建快捷菜单	235
10.3 用编程方式定义菜单	236
10.3.1 条形菜单定义	236
10.3.2 弹出式菜单设计	237
10.4 创建 SDI 菜单	240
习题 10	241
上机操作 10	242

第 11 章 报表和标签设计 243

11.1 创建报表	243
11.1.1 报表布局	243
11.1.2 使用报表向导创建报表	243
11.1.3 使用报表设计器创建报表	247
11.1.4 使用快速报表命令创建报表	247

11.2 报表设计	248	13.2 为运行环境准备应用程序	280
11.2.1 设置数据环境	249	13.2.1 选择连编类型	280
11.2.2 设计报表布局	249	13.2.2 应用程序的最小运行环境	281
11.2.3 添加报表控件	251	13.2.3 确保应用程序正确运行	281
11.3 数据分组和多栏报表	255	13.2.4 在应用程序中使用资源文件	282
11.3.1 设计分组报表	256	13.2.5 删除应用程序中受限制的功能 和文件	283
11.3.2 设计多栏报表	260	13.3 定制要发布的程序	285
11.4 创建标签	261	13.3.1 保护源代码并将其存档	285
11.4.1 使用标签向导创建标签	262	13.3.2 处理应用程序的运行错误	286
11.4.2 使用标签设计器创建标签	263	13.3.3 给应用程序添加帮助信息	286
11.5 打印报表和标签	264	13.3.4 修改应用程序的外观	287
习题 11	265	13.3.5 备份应用程序的源代码	287
上机操作 11	266	13.3.6 生成可发布的应用程序	288
第 12 章 导入与导出数据	269	13.4 创建发布盘	288
12.1 导入外部数据	269	13.4.1 创建发布盘的步骤	288
12.1.1 使用导入向导导入外部数据	269	13.4.2 再论发布树和安装向导	289
12.1.2 使用命令导入外部数据	273	13.4.3 利用安装向导创建发布盘	290
12.2 导出数据	274	习题 13	292
12.2.1 使用菜单导出数据	274	上机操作 13	292
12.2.2 使用 EXPORT 命令导出数据	276	附录 模拟测试题	293
12.3 与其他应用程序共享信息	277	模拟测试题 1	293
12.3.1 复制和粘贴数据	277	模拟测试题 2	297
12.3.2 创建邮件合并	278	模拟测试题 3	301
习题 12	278	模拟测试题 4	305
上机操作 12	279	模拟测试题 5	309
第 13 章 发布应用程序	280		
13.1 应用程序的发布过程	280		

第1章 Visual FoxPro 基础

Visual FoxPro(简称 VFP) 6.0 是 Microsoft 公司推出的一个优秀的可视化数据库管理系统。使用它, 用户不仅可以创建和管理数据库, 而且可以设计各种数据库应用程序。

1.1 数据库基础知识

数据库技术是在 20 世纪 60 年代末兴起的一项数据管理技术。数据库是按照一定方式把相关数据组织、存储在计算机中的数据的集合。数据库不仅存放数据, 而且还存放数据之间的联系。

1.1.1 数据库的发展

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) 人工管理阶段

20 世纪 50 年代中期, 外存储器只有纸带、磁带、卡片等, 没有像磁盘这样的速度快、存储容量大、随机访问、直接存储的外存储器。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。这一阶段的数据管理称为人工管理阶段。

(2) 文件系统阶段

20 世纪 50 年代末期, 计算机开始大量地用于管理中的数据处理工作。在硬件方面, 磁盘成为主要的外存; 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数据的软件。

在文件系统阶段, 程序与数据有了一定的独立性, 程序和数据是分开存储的。数据文件可被多次存取。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于数据处理的算法上, 而不必关心记录在存储器上的地址和内存外存交换数据的过程。这一阶段的数据管理称为文件管理阶段。

(3) 数据库系统阶段

随着社会信息量的迅速增长, 计算机处理的数据量不断增加, 文件管理系统采用的一次最多存取一个记录的访问方式, 以及在不同文件之间缺乏相互联系的结构, 越来越不能适应大量数据管理的需要。于是数据库管理系统便应运而生, 并在 20 世纪 60 年代末期诞生了第一个商品化的数据库系统——美国 IBM 公司的 IMS 系统。

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,从主机—终端体系结构发展到客户/服务器系统结构。

(5) 面向对象数据库系统

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,它的发展十分迅猛,其影响涉及计算机科学及其应用的各个领域。

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。

1.1.2 数据库系统

1.1.2.1 基本概念

(1) 数据库

数据库(DB)是将许多具有相关性的数据以一定的方式编排存储在一起,形成的科学化数据集合。换句话说,数据库不只是一个存放数据的“仓库”,而是一个通用化的综合性数据集合,是把众多的数据按一定的结构,有联系地组织在一起。数据库不仅存放数据,而且存放数据之间的关系。

(2) 数据库管理系统

为了让多种应用程序并发地使用数据库中具有最小冗余度的共享数据,必须使数据与程序具有较高的独立性。这就需要有一个软件系统对数据实行专门管理,提供安全性和完整性等统一控制机制,方便用户以交互式命令或程序方式对数据库进行操作。

数据库管理系统(DBMS)就是对数据库的建立、使用和维护进行管理的软件系统。数据库管理系统主要包括三部分:数据描述语言(DDL),数据操作语言(DML)以及管理和控制程序。

(3) 数据库应用系统

数据库应用系统(DBAS)是指系统开发人员利用数据库系统资源开发出来的,面向某一实际应用的应用软件系统。一个 DBAS 通常由数据库和应用程序两部分构成,它们都需要在数据库管理系统支持下开发和工作。

(4) 数据库系统

数据库系统是指引进数据库技术后的计算机系统,实现有组织地、动态地存储大量相关数据,提供数据处理和信息资源共享的便利手段。数据库系统由四部分组成(见图 1.1):计算机系统、数据库、数据库管理系统及有关人员。

- 计算机系统包括硬件系统及相关的软件如操作系统。
- 数据库是指数据库系统包含的若干个设计合理、满足应用需要的数据库。
- 数据库管理系统是指对数据库中的数据进行管理的软件。
- 有关人员包括数据库管理员和用户。



图 1.1 数据库系统层次示意图

1.1.2.2 数据库系统的特点

与文件系统比较，数据库系统有下列特点。

(1) 数据结构化

在文件系统中，各个文件不存在相互联系；而数据库系统则不同，在同一数据库中的数据文件是有联系的，且在整体上服从一定的结构形式。

(2) 数据共享

共享是数据库系统的目的，也是它的重要特点。而在文件系统中，数据一般是专用的。

(3) 数据独立性

在文件系统中，数据结构和应用程序相互依赖，一方的改变总是要影响另一方。数据库系统则较大程度地改善了这种情况，减少了数据和程序之间的相互依赖。

(4) 可控冗余度

文件系统中，每个用户拥有并使用自己的数据，难免有许多数据相互重复，这就是冗余。实现共享后，不必要的重复将全部消除，但为了提高查询效率，有时也保留少量重复数据，其冗余度可由设计人员控制。

1.1.3 数据模型

通常把表示客观事物及其联系的数据及结构称为数据模型。

1.1.3.1 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等等。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，也可能是抽象的事件。如教师、图书等属于实际事物；借阅图书、授课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，如教师实体用(专业，姓名，性别，职务，学历)等若干个属性来描述；图书实体用(分类号，书名，作者，单价)等多个属性来描述。

(3) 实体集和实体型

属性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,在教师实体集当中,(计算机,李明,男,68/12/03,教授)表示教工名册中的一个具体的人;在图书实体集中,(TP322,数据库基础与应用,马刚,29.80)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:教师档案表、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。

1.1.3.2 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。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少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

学院和院长两个实体型,一个学院只有一个院长,而一个院长不能同时在其他学院兼任院长,在这种情况下,学院和院长之间存在一对一的联系。

在 VFP 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。如,某系一个班学生档案表和教务处存档的某科成绩表之间就存在一对一的联系。

(2) 一对多联系

系部和学生两个实体型,一个系有很多个学生,而一个学生只能在一个系里注册,系和学生是一对多的联系。

在 VFP 中,一对多的联系表现为主表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。

(3) 多对多联系

学生和课程两个实体型,一个学生可以选修多门课程,一门课程可以由多个学生选修。因此,学生和课程间存在多对多的联系。

在 VFP 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。

1.1.3.3 数据模型简介

数据库中的数据是按照一定的结构组织和存储的,这种结构用数据模型来表示。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。

(1) 层次模型

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,数据的组织是由“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 给出了一个层次模型示例。



图 1.2 层次模型示例

(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。图 1.3 是一个简单的网状模型。



图 1.3 网状模型示例

(3) 关系模型

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。

关系模型与层次模型、网状模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。

1.1.4 关系数据库

自 20 世纪 80 年代以来,新推出的数据库管理系统几乎都支持关系模型,Visual FoxPro 就是一种关系数据库管理系统。下面将结合 Visual FoxPro 来介绍关系数据库系统的基本概念。

1.1.4.1 关系模型中的常用术语

(1) 关系

关系数据库是依据关系模型建造的数据库,方法是把一些复杂的数据结构归结为简单的二维表形式,一张二维表就是一个关系,每个关系有一个关系名。在 VFP 中,一个关系存储为一个文件,称为表,文件扩展名为 .dbf。图 1.4 给出一个学生表和一个成绩表两个关系。在这两个表中都有惟一标识一个学生的属性——学号,根据学号通过一定的关系运算可以把两个关系联系起来。

(2) 元组(记录)

在一个二维表中,水平方向的行称为元组,每一行是一个元组。元组对应存储文件中的

学号	姓名	性别	出生日期	四级通过	奖学金	总成绩
19980101	李小飞	男	07/08/79	T	400.0	0.0
19980201	王美英	女	05/01/80	T	500.0	0.0
19980102	张欣	女	09/20/80	F	200.0	0.0
19980303	王--林	男	08/06/81	T	500.0	0.0
19980204	赵玉	女	08/02/78	F	300.0	0.0

学号	英语	数学	计算机
19980201	89.0	76.0	80.0
19980102	80.0	85.0	75.0
19980303	80.0	80.0	92.0
19980101	70.0	82.0	79.0
19980301	91.0	90.0	89.0

图 1.4 学生表和成绩表

一个具体记录。如学生表和成绩表两个关系各包括多条记录(或多个元组)。

(3) 属性(字段)

二维表中垂直方向的列称为属性, 每一列有一个属性名, 在 VFP 中表示为字段名。

(4) 域

属性的取值范围, 即不同元组对同一个属性的取值所限定的范围。

(5) 关键字

属性或属性的组合, 其值能够惟一地标识一个元组。在 VFP 中表示为字段或字段的组合。如学生表中的学号可以作为标识一条记录的关键字。而性别因为男的女的都不只一个人, 因此性别就不能作为关键字。

(6) 外部关键字

如果表中的一个字段不是本表的主关键字或候选关键字, 而是另外一个表的主关键字或候选关键字, 这个字段就称为外部关键字。

1.1.4.2 关系的特点

关系模型看起来简单, 但是并不能把日常手工管理所用的各种表格, 按照一张表一个关系直接存放到数据库系统中。在关系模型中对关系有一定的要求, 关系必须具有以下特点。

- ① 每个属性必须是不可分割的数据单元, 即表中不能再包含表。
- ② 在同一个关系中不能出现相同的属性名。
- ③ 关系中不允许有完全相同的元组, 即冗余。
- ④ 在一个关系中行和列的次序无关紧要。

1.1.4.3 关系运算

对关系数据库进行查询时, 要找到需求的数据, 就要对关系进行一定的关系运算。关系的基本运算有两类, 一类是传统的集合运算(并、交、差等), 另一类是专门的关系运算(选择、投影、联接)。

(1) 传统的集合运算

进行并、差、交集运算的两个关系必须具有相同的模式, 即相同结构。

① 并。两个相同结构关系的并是由属于这两个关系的元组组成的集合。如, 有两个结构相同的学生关系 R1, R2, 分别存放两个班的学生, 把第二个班的学生记录追加到第一个班的学生记录后面就是这两个关系的并集。

② 差。设有两个相同结构的关系 R 和 S, R 差 S 的结果是由属于 R 但不属于 S 的元组组成的集合, 即差运算的结果是从 R 中去掉 S 中也有的元组。

例如, 有选学 VFP 课程的学生关系 R, 选学 VB 课程的学生关系 S, 求选学 VFP 课程而没选学 VB 课程的学生, 就应当进行差运算。

③ 交。两个具有相同结构的关系 R 和 S, 它们的交是既属于 R 又属于 S 的元组组成的集合。交运算的结果是 R 和 S 的共同元组。

例如, 有选学 VFP 课程的学生关系 R, 选学 VB 课程的学生关系 S, 求既选 VFP 课程又选 VB 课程的学生, 就应当进行交运算。

在 VFP 中没有直接提供传统的集合运算, 可通过其他操作或编写程序来实现。

(2) 专门的关系运算

在 VFP 中, 查询是高度非过程化的, 用户只需明确提出“要做什么”, 而不需要指出“怎么做”。系统将自动对查询过程进行优化, 可以实现对多个相关联的表的高速存取。然而, 要正确表示复杂的查询并非是一件简单的事, 了解专门的关系运算有利于正确给出查询表达式。

① 选择。从关系中找出满足给定条件的元组的操作就称为选择。例如要从“成绩表”中找出计算机成绩大于 85 分的学生, 所进行的查询操作就属于选择运算。

选择是从行的角度进行的运算, 即从水平方向抽取记录。经过选择运算得到的结果可以形成新的关系, 其关系模式不变, 但其中的元组是原关系的一个子集。

② 投影。从关系模式中指定若干个属性组成新的关系称为投影。

投影是从列的角度进行的运算。经过投影运算可以得到一个新的关系, 其关系模式所包含的属性个数往往比原关系少, 或者属性的排列顺序不同。例如从“学生表”中查询学生姓名及四级通过情况所进行的查询操作就属于投影运算。

③ 联接。联接是关系的横向结合, 联接运算将两个关系模式拼接成一个更宽的关系模式, 生成的新关系中包含满足联接条件的元组。

联接过程是通过联接条件来控制的, 联接条件中将出现两个表中的公共属性名, 或者具有相同语义、可比的属性。联接结果是满足条件的所有记录。

选择和投影的操作对象只是一个表, 相当于对一个二维表进行切割。联接运算需要两个表作为操作对象。如果需要联接两个以上的表, 应当两两进行联接。

1.2 Visual FoxPro 6.0 系统概述

1995 年, 微软公司成功地将用户使用多年的 FoxPro 数据库发展成支持可视化开发和操作环境的面向对象的数据库——Visual FoxPro 3.0, 该产品获得巨大成功。随后微软公司推出了 Visual FoxPro 5.0, 1998 年, 微软公司又推出了 Visual FoxPro 6.0, 该产品同 Visual C++ 6.0, Visual Basic 6.0 等产品共同组成 Microsoft Visual Studio 98。Visual FoxPro 6.0 中文版是 Visual FoxPro 6.0 英文版的汉化版本, 主要是为了面向中国市场。

1.2.1 Visual FoxPro 6.0 的特点

Visual FoxPro 6.0 中文版在以前版本的基础上增强了很多功能, 主要表现在向微机系列开发平台的标准靠拢和引进, 对 Internet 和 Intranet 提供更加强大的支持。

(1) 强大的查询与管理功能

VFP 拥有近 500 条命令和 200 余种函数, 其功能空前强大。

由于采用了 Rushmore 快速查询技术, VFP 能从具有众多记录的数据库表中迅速找出一组满足查询要求的记录。

VFP 提供了一种称为项目管理器的管理工具, 可供用户对所开发项目中的数据、文档、源代码和类库等资源集中进行高效的管理。

(2) 大量使用可视化的界面操作工具

VFP 6.0 可提供向导、设计器、生成器等操作工具, 帮助用户以简单的操作快速完成各种查询和设计任务。

(3) 支持面向对象的程序设计

既使用结构化程序设计, 也应用面向对象的程序设计, 构成了 VFP 程序设计的特点, 同时为用户编程带来了很大的方便。

(4) 扩大了对 SQL 语言的支持

SQL 语言是关系数据库的标准语言, 其查询语句不仅功能强大, 而且使用灵活。这不仅加强了 VFP 语言的功能, 也为用户提供了学习与熟悉 SQL 语言的机会。

(5) 通过 OLE 实现应用集成

“对象链接与嵌入”(简称 OLE)是微软公司开发的一项重要技术。通过这种技术, VFP 可与微软公司其他应用软件共享数据, 实现应用集成。比如在不退出 VFP 环境的情况下, 用户就可在 VFP 的表单(或窗体)中链接其他软件中的对象, 直接对这些对象进行编辑。在通过必要的格式转换后, 用户可以在 VFP 与其他软件之间进行数据的输入与输出。

(6) 支持网络应用

VFP 既适用于单机环境, 也适用于网络环境。其网络功能主要包括:

- ① 支持客户/服务器结构。既可访问本地计算机, 也支持对服务器的浏览。
- ② 对于来自本地、远程或多个数据库表的异种数据, VFP 可支持用户通过本地或远程视图访问与使用, 并在需要时更新表中的数据。
- ③ 在多用户环境下, VFP 还允许建立事务处理程序来控制对数据的共享, 包括支持用户共享数据, 或限制部分用户访问某些数据等。

1.2.2 Visual FoxPro 6.0 的安装与启动

1.2.2.1 VFP 系统环境

Visual FoxPro 6.0 的功能强大, 但是它对系统的要求并不高, 个人计算机的软硬件基本配置要求如下:

- ① 处理器: 带有 486DX/66 MHz 处理器, 推荐使用 Pentium 或更高档次的处理器。
- ② 内存: 16MB 以上的内存。
- ③ 硬盘空间: 典型安装需要 85MB 的硬盘空间; 最大安装需要 90MB 硬盘空间。
- ④ 操作系统: Windows 95/98(中文版)或更高版本的操作系统。

1.2.2.2 安装 Visual FoxPro 6.0

(1) 单机安装

- ① 将光盘插入 CD-ROM 驱动器。
- ② 在“我的电脑”或“资源管理器”中选择光驱, 找到 setup.exe 文件并运行。
- ③ 按照安装向导, 选择安装形式并完成安装。
- ④ 退出安装。

(2) 网络安装

网络上的用户,可实现资源共享,安装步骤如下:

- ① 将光盘插入与网络相连的任何共享 CD-ROM 驱动器中。
- ② 在“资源管理器”中选择“映射网络驱动器”将 CD-ROM 进行驱动器映射。
- ③ 在“我的电脑”或“资源管理器”中选择光驱,找到 setup.exe 文件并运行。
- ④ 按照安装向导,选择安装形式完成安装后退出安装。

1.2.2.3 启动 VFP

启动 Visual FoxPro 6.0 的方法很多,如利用开始菜单启动:

单击【开始】按钮,执行程序选项中的 Microsoft Visual FoxPro 6.0 即可启动 VFP。

1.2.2.4 退出 VFP

退出 VFP 有下面一些方法:

- ① 在命令窗口中输入“quit”命令按回车键;
- ② 激活 VFP 窗口,按 Alt + F4 组合键;
- ③ 双击主菜单窗口左上角的控制图标;
- ④ 单击主菜单窗口左上角的控制图标,在弹出的菜单中选择【关闭】命令;
- ⑤ 执行主窗口【文件】菜单中的【退出】命令;
- ⑥ 右单击任务栏的窗口标题,在弹出的菜单中选择【关闭】命令;
- ⑦ 单击标题栏上的【关闭】按钮。

1.2.3 Visual FoxPro 6.0 的用户界面

系统启动成功后,进入如图 1.5 所示的主界面。

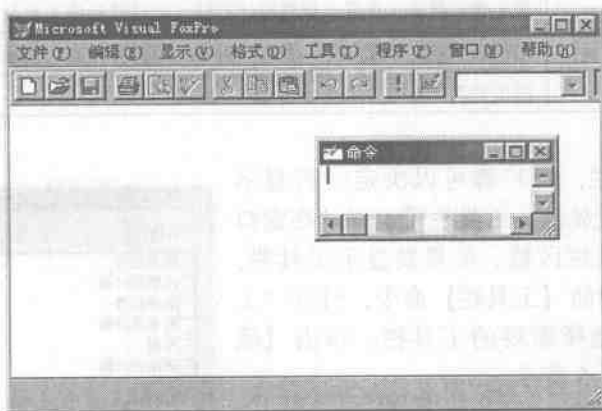


图 1.5 Visual FoxPro 6.0 的主界面

1.2.3.1 主窗口界面

进入主窗口以后,用户可以根据自己的需要进行各种操作。主窗口主要由标题栏、菜单栏、工具栏、状态栏和命令窗口组成。

- ① 标题栏:显示目前所使用的系统是 Microsoft Visual FoxPro。
- ② 菜单栏:提供了如“文件”、“编辑”、“视图”、“插入”、“格式”、“工具”、“表格”、“窗口”和“帮助”等菜单,每个菜单都由一组命令组成,应用程序的开发可通过菜单中的这些命令实现。