

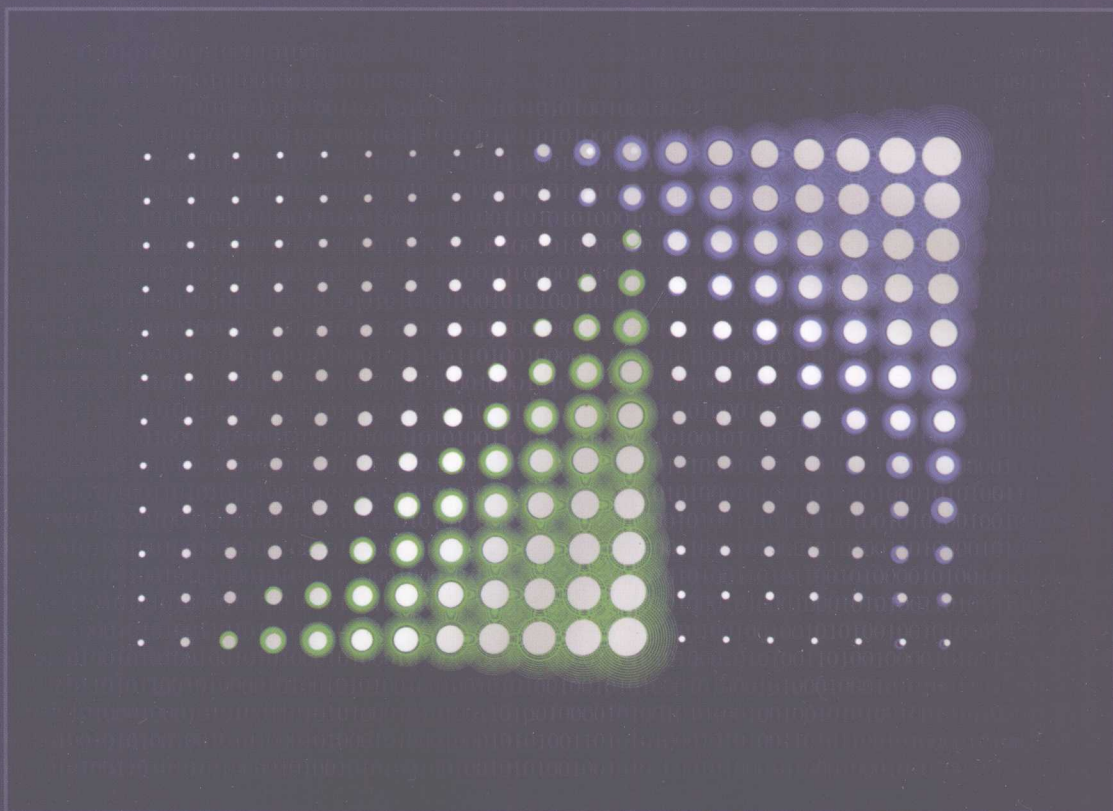


北京市高等教育精品教材立项项目

新编计算机类本科规划教材

# Visual FoxPro 程序设计实用教程

刘 丽 金晓龙 编著



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

北京市高等教育精品教材立项项目

新编计算机类本科规划教材

# Visual FoxPro 程序设计实用教程

刘 丽 金晓龙 编著

电子工业出版社

Publishing House of Electronics Industry

北京 · BEIJING

## 内 容 简 介

本书按照项目驱动的方式,以知识点和实例为主线介绍了 Visual FoxPro 的各种基本操作,并将一个系统开发实例贯穿到每个章节中,最终完成该系统的开发。本教材共分为 10 章,分别介绍了数据库系统及 Visual FoxPro 的基本概念、项目管理器及其操作、数据表的基本操作、数据库的设计与操作、程序设计基础、面向对象程序设计、表单的建立与使用、结构化查询语言(SQL)、查询与视图、报表设计、菜单设计、应用程序的生成和发布等,并配有丰富的例题、习题。本书不仅给出了系统设计步骤和源代码,而且突出介绍了各部分的设计思路、实现目标和设计流程,力求使读者抓住本质,迅速掌握使用 Visual FoxPro 开发管理信息系统的相关知识与技术。

本书较好地在学习与实践结合在一起,既可作为高等院校非计算机专业应用型本科学生的教材,也可作为希望尽快开发小型数据库应用程序的技术人员的参考书。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

### 图书在版编目(CIP)数据

Visual FoxPro 程序设计实用教程 / 刘丽, 金晓龙编著. —北京: 电子工业出版社, 2009.6

新编计算机类本科规划教材

ISBN 978-7-121-08686-1

I. V… II. ①刘…②金… III. 关系数据库—数据库管理系统, Visual FoxPro 6.0—程序设计—高等学校—教材 IV. TP311.138

中国版本图书馆 CIP 数据核字(2009)第 061057 号

策划编辑: 冯小贝

责任编辑: 冯小贝

印 刷: 北京季蜂印刷有限公司

装 订: 三河市万和装订厂

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×1092 1/16 ·印张: 19.5 字数: 499 千字

印 次: 2009 年 6 月第 1 次印刷

定 价: 29.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010) 88258888。

# 前 言

在数据库技术飞速发展的今天, Visual FoxPro 作为优秀的新一代小型数据库管理系统, 仍然受到众多用户的青睐。Visual FoxPro 具有操作界面友好、功能完善和强大、语言简练、简单易学、辅助开发工具丰富、便于实现应用系统快速开发等特点, 深受许多小型数据库应用系统开发人员的喜爱。为了满足高等学校 Visual FoxPro 程序设计类课程的教学需要, 我们编写了这本《Visual FoxPro 程序设计实用教程》。作为北京市精品教材立项项目, 在项目实施的过程中, 项目组成员为了能够编写出优质的教材做出了许多努力。在编写本书的过程中, 依据我们多年的实际教学经验, 并且参考和借鉴了多本相关的同类教材, 将一个系统开发实例贯穿到每个章节中, 并最终完成该系统的开发。通过对各个具体实例的精心解析, 读者不仅能够全面了解 Visual FoxPro 的基本风格, 学会基本操作技能, 同时还能从其各具示范性的操作过程中获得有益的启示, 积累自己的开发经验。本书不仅给出了系统设计步骤和源代码, 而且突出介绍了各部分的设计思路、实现目标和设计流程, 力求使读者抓住本质, 迅速掌握使用 Visual FoxPro 开发管理信息系统的相关知识, 学到设计管理信息系统的思路和方法。本书通过大量的实例, 介绍在可视环境下进行面向对象程序设计的方法、步骤, 力求通过实际操作让读者熟悉 Visual FoxPro 的使用方法。通过一个个具有针对性的实例和具体的操作过程, 让读者很快熟悉 Visual FoxPro 的应用环境, 从而实现从易到难, 边学、边练、边提高的学习过程。

本书以 Visual FoxPro 6.0 中文版为平台, 结合普通高校非计算机专业数据库程序设计课程的具体要求, 深入浅出地介绍了 Visual FoxPro 数据库程序设计的有关知识、方法和具体的实例。本教材共分 10 章, 分别介绍了数据库系统及 Visual FoxPro 的基本概念、项目管理器及其操作、数据表的基本操作、数据库的设计与操作、程序设计基础、面向对象程序设计、表单的建立与使用、结构化查询语言 (SQL)、查询与视图、报表设计、菜单设计、应用程序的生成和发布等。

本书以基础知识、基本理论和基本方法为着眼点, 努力做到知识体系完整, 结构顺序合理, 内容深度适宜, 例题典型全面, 讲解深入浅出, 使用起来得心应手。在最后一章中给出了应用系统开发的具体案例, 通过该案例可使读者更加详细地了解 and 掌握应用系统开发的一般步骤、基本方法和具体过程, 使读者快速获得应用系统开发的基本技能。

本书的风格简洁实用, 内容贴近实际应用, 是一本较好地将学习与实践结合在一起的书籍。本书内容翔实、案例完整、步骤详尽, 具有很强的实用性, 十分适合那些希望尽快实现应用开发的读者的需要。

由于作者水平有限, 加上编写时间仓促, 错误和不足之处在所难免, 敬请广大读者批评指正。

# 目 录

|                                 |      |
|---------------------------------|------|
| 第 1 章 Visual FoxPro 基础知识        | (1)  |
| 1.1 数据库基础概念                     | (1)  |
| 1.1.1 数据库相关知识                   | (1)  |
| 1.1.2 数据模型                      | (2)  |
| 1.1.3 关系模型的基本概念                 | (4)  |
| 1.1.4 关系运算                      | (6)  |
| 1.1.5 实体模型                      | (7)  |
| 1.2 Visual FoxPro 6.0 及其界面      | (8)  |
| 1.2.1 Visual FoxPro 6.0 概述      | (8)  |
| 1.2.2 Visual FoxPro 6.0 的启动与退出  | (9)  |
| 1.2.3 Visual FoxPro 6.0 系统的主窗口  | (10) |
| 1.3 项目管理器                       | (12) |
| 1.3.1 【项目管理器】窗口的组成              | (12) |
| 1.3.2 建立项目文件                    | (13) |
| 1.3.3 使用项目管理器管理项目文件             | (14) |
| 1.3.4 操作【项目管理器】窗口               | (15) |
| 1.3.5 Visual FoxPro 6.0 的辅助设计工具 | (16) |
| 1.4 学生成绩管理系统案例分析                | (17) |
| 1.4.1 案例设计分析                    | (17) |
| 1.4.2 学生成绩管理系统功能模块设计            | (18) |
| 1.4.3 学生成绩管理系统设计要求              | (23) |
| 思考与练习 1                         | (23) |
| 第 2 章 数据与数据运算                   | (26) |
| 2.1 数据与运算                       | (26) |
| 2.1.1 数据类型                      | (26) |
| 2.1.2 常量和变量                     | (26) |
| 2.1.3 运算符及表达式                   | (28) |
| 2.2 常用函数                        | (31) |
| 2.2.1 数学运算函数                    | (32) |
| 2.2.2 字符处理函数                    | (33) |
| 2.2.3 日期和时间函数                   | (35) |
| 2.2.4 测试函数                      | (35) |

|                                        |             |
|----------------------------------------|-------------|
| 2.2.5 转换函数 .....                       | (37)        |
| 2.2.6 用户自定义对话框 MESSAGEBOX()函数的使用 ..... | (38)        |
| 思考与练习 2 .....                          | (41)        |
| <b>第 3 章 数据库的基本操作 .....</b>            | <b>(43)</b> |
| 3.1 数据库与数据表 .....                      | (43)        |
| 3.1.1 数据库的基本操作 .....                   | (43)        |
| 3.1.2 数据表的基本概念 .....                   | (45)        |
| 3.1.3 数据表结构的建立 .....                   | (47)        |
| 3.1.4 数据表结构的操作 .....                   | (50)        |
| 3.1.5 数据表记录的操作 .....                   | (53)        |
| 3.2 数据表的索引与筛选 .....                    | (62)        |
| 3.2.1 建立索引 .....                       | (62)        |
| 3.2.2 筛选记录 .....                       | (65)        |
| 3.3 Visual FoxPro 中的命令操作 .....         | (66)        |
| 3.3.1 Visual FoxPro 命令操作的基本概念 .....    | (66)        |
| 3.3.2 数据库、数据表的操作命令 .....               | (70)        |
| 3.3.3 表记录的操作命令 .....                   | (71)        |
| 3.3.4 追加记录命令 .....                     | (73)        |
| 3.3.5 记录的排序和索引 .....                   | (74)        |
| 3.3.6 记录的查询 .....                      | (75)        |
| 3.3.7 数值统计 .....                       | (76)        |
| 3.4 多表操作 .....                         | (78)        |
| 3.4.1 数据表的关联 .....                     | (78)        |
| 3.4.2 设置参照完整性 .....                    | (80)        |
| 思考与练习 3 .....                          | (83)        |
| <b>第 4 章 查询与视图 .....</b>               | <b>(85)</b> |
| 4.1 查询 .....                           | (85)        |
| 4.1.1 查询文件的建立与修改 .....                 | (85)        |
| 4.1.2 查询文件的执行 .....                    | (91)        |
| 4.2 视图 .....                           | (98)        |
| 4.2.1 利用向导创建本地视图 .....                 | (99)        |
| 4.2.2 利用视图设计器创建本地视图 .....              | (100)       |
| 4.2.3 创建参数化视图 .....                    | (104)       |
| 4.2.4 创建远程视图 .....                     | (105)       |
| 4.3 关系数据库标准语言 SQL .....                | (108)       |
| 4.3.1 SQL 概述 .....                     | (108)       |
| 4.3.2 SQL 的各类查询 .....                  | (110)       |
| 4.3.3 操作功能 .....                       | (125)       |

|              |                                   |       |
|--------------|-----------------------------------|-------|
| 4.3.4        | 定义功能 .....                        | (126) |
| 4.3.5        | 定义视图 .....                        | (130) |
| 思考与练习 4      | .....                             | (134) |
| <b>第 5 章</b> | <b>Visual FoxPro 程序设计基础</b> ..... | (136) |
| 5.1          | 程序设计基础 .....                      | (136) |
| 5.1.1        | 程序的建立与维护.....                     | (136) |
| 5.1.2        | 编程基础 .....                        | (138) |
| 5.2          | 程序的流程控制 .....                     | (141) |
| 5.2.1        | 选择结构 .....                        | (141) |
| 5.2.2        | 循环结构 .....                        | (146) |
| 5.3          | 模块结构程序设计 .....                    | (151) |
| 5.3.1        | 过程及过程调用 .....                     | (151) |
| 5.3.2        | 过程调用中的参数传递 .....                  | (154) |
| 5.3.3        | 变量的作用域 .....                      | (156) |
| 5.3.4        | 自定义函数 .....                       | (157) |
| 思考与练习 5      | .....                             | (161) |
| <b>第 6 章</b> | <b>表单</b> .....                   | (165) |
| 6.1          | 面向对象程序设计简介 .....                  | (165) |
| 6.1.1        | 面向对象程序设计中的基本概念 .....              | (165) |
| 6.1.2        | Visual FoxPro 中的类 .....           | (166) |
| 6.1.3        | Visual FoxPro 中的对象 .....          | (168) |
| 6.1.4        | Visual FoxPro 中的事件和事件过程.....      | (171) |
| 6.2          | 表单的创建 .....                       | (174) |
| 6.2.1        | 使用表单向导创建表单 .....                  | (174) |
| 6.2.2        | 使用表单设计器创建表单 .....                 | (178) |
| 6.2.3        | 表单控件 .....                        | (181) |
| 6.3          | 常用控件的使用方法 .....                   | (185) |
| 6.3.1        | 标签、文本框和命令按钮(组)的使用 .....           | (185) |
| 6.3.2        | 编辑框、列表框和组合框的使用 .....              | (190) |
| 6.3.3        | 选项按钮组、复选框和微调按钮的使用 .....           | (192) |
| 6.3.4        | 表格和计时器的使用 .....                   | (196) |
| 6.3.5        | OLE 绑定控件和图像的使用 .....              | (198) |
| 思考与练习 6      | .....                             | (206) |
| <b>第 7 章</b> | <b>菜单与工具栏</b> .....               | (208) |
| 7.1          | 菜单的建立 .....                       | (208) |
| 7.1.1        | 菜单概述 .....                        | (208) |
| 7.1.2        | 创建快速菜单 .....                      | (211) |

|               |                             |              |
|---------------|-----------------------------|--------------|
| 7.1.3         | 创建菜单 .....                  | (212)        |
| 7.1.4         | 快捷菜单的建立 .....               | (218)        |
| 7.1.5         | 设置对菜单项的响应 .....             | (219)        |
| 7.2           | 工具栏 .....                   | (224)        |
| 7.2.1         | 定制与建立工具栏 .....              | (224)        |
| 7.2.2         | 给新工具栏命名 .....               | (226)        |
| 7.2.3         | 定义对象操作 .....                | (226)        |
|               | 思考与练习 7 .....               | (228)        |
| <b>第 8 章</b>  | <b>报表与标签 .....</b>          | <b>(230)</b> |
| 8.1           | 报表的设计 .....                 | (230)        |
| 8.1.1         | 报表类型 .....                  | (230)        |
| 8.1.2         | 利用报表向导创建报表 .....            | (230)        |
| 8.1.3         | 使用设计器创建报表 .....             | (235)        |
| 8.1.4         | 对报表进行细节设计 .....             | (237)        |
| 8.1.5         | 完善报表 .....                  | (242)        |
| 8.2           | 创建标签 .....                  | (245)        |
| 8.2.1         | 使用标签向导创建标签 .....            | (245)        |
| 8.2.2         | 利用标签设计器创建标签 .....           | (247)        |
| 8.3           | 报表和标签的输出 .....              | (249)        |
| 8.3.1         | 页面设置 .....                  | (249)        |
| 8.3.2         | 打印输出报表和标签 .....             | (250)        |
|               | 思考与练习 8 .....               | (251)        |
| <b>第 9 章</b>  | <b>应用程序的编译与发布 .....</b>     | <b>(253)</b> |
| 9.1           | 应用程序的开发 .....               | (253)        |
| 9.1.1         | 系统开发步骤 .....                | (253)        |
| 9.1.2         | 连编应用程序 .....                | (254)        |
| 9.1.3         | 主程序设计 .....                 | (256)        |
| 9.2           | 应用程序生成器 .....               | (259)        |
| 9.2.1         | 使用应用程序向导 .....              | (259)        |
| 9.2.2         | 应用程序生成器 .....               | (260)        |
| 9.2.3         | 建立 Visual FoxPro 应用程序 ..... | (262)        |
| 9.2.4         | 制作安装盘 .....                 | (266)        |
|               | 思考与练习 9 .....               | (270)        |
| <b>第 10 章</b> | <b>小型系统项目开发 .....</b>       | <b>(272)</b> |
| 10.1          | 教职工信息管理系统 .....             | (272)        |
| 10.1.1        | 案例设计的提出及要解决的问题 .....        | (272)        |
| 10.1.2        | 案例设计要实现的主要功能 .....          | (272)        |

|        |                      |       |
|--------|----------------------|-------|
| 10.1.3 | 案例设计操作思路 .....       | (273) |
| 10.1.4 | 案例设计参考步骤 .....       | (273) |
| 10.2   | 图书信息管理系统 .....       | (285) |
| 10.2.1 | 案例设计的提出及要解决的问题 ..... | (285) |
| 10.2.2 | 案例设计要实现的主要功能 .....   | (285) |
| 10.2.3 | 案例设计操作思路 .....       | (285) |
| 10.2.4 | 案例设计参考步骤 .....       | (285) |
|        | 思考与练习 10 .....       | (299) |
|        | 参考文献 .....           | (301) |

# 第 1 章 Visual FoxPro 基础知识

## 1.1 数据库基础概念

数据库管理技术是信息科学的重要组成部分。随着商品经济的发展、科学技术的进步和激烈的市场竞争,社会信息量倍增,决策难度也随之加大,使得计算机处理的数据量不断增加。于是数据库管理系统便应运而生,从而也促进了信息科学的发展。下面我们从数据库基础概念开始介绍。

### 1.1.1 数据库相关知识

#### 1. 数据、信息

① 数据 (data): 是对客观事物特征所进行的一种抽象化、符号化的表示。通俗地讲,凡是能被计算机接受,并能被计算机处理的数字、字符、图形、声音、图像等统称为数据。数据所反映的事物属性是它的内容,而符号是它的形式。

② 信息 (information): 是客观事物属性的反映。它所反映的是关于某一客观系统中某一事物的某一方面属性或某一时刻的表现形式。通俗地讲,信息是经过加工处理并对人类客观行为产生影响的数据表现形式。也可以说,信息是有一定含义的并经过加工处理,能够提供决策性依据的数据。

#### 2. 数据处理

所谓数据处理实际上就是利用计算机对各种类型的数据进行处理。它包括对数据的采集、整理、存储、分类、排序、检索、维护、加工、统计和传输等一系列操作过程。数据处理的目的是从大量的、原始的数据中获得我们所需要的资料并提取有用的数据成分,作为行为和决策的依据。

#### 3. 数据库

数据库在英语中称为 database。分解这个单词来看, data 是数据, base 可译为基地或仓库。所以在通俗的意义上,不妨将数据库理解为存储数据的基地。它是以一定的组织方式将相关的数据组织在一起并存储在外部存储器上,所形成的能为多个用户共享的、与应用程序彼此独立的一组相互关联的数据集合。

#### 4. 数据库管理系统

数据库管理系统简称 DBMS (Data Base Management System)。DBMS 是操纵和管理数据库的软件,是数据库系统的管理控制中心,一般有 4 大功能: 数据定义功能、数据库操作功能、控制和管理功能、建立和维护功能。

## 5. 数据库系统

把以数据库应用为基础的计算机系统称为数据库系统。它是一个实际可行的并按照数据库方式存储、维护和管理数据的系统。数据库系统通常由计算机硬件、数据库、数据库管理系统、相关软件、人员（数据库管理员、应用程序员、用户）等组成，如图 1.1 所示。

## 6. 数据库应用系统

数据库应用系统是一个复杂的系统，它由硬件、操作系统、数据库管理系统、编译系统、用户应用程序和数据库组成。

数据库、数据库管理系统和数据库系统是 3 个不同的概念，数据库管理系统在计算机中的地位如图 1.2 所示。

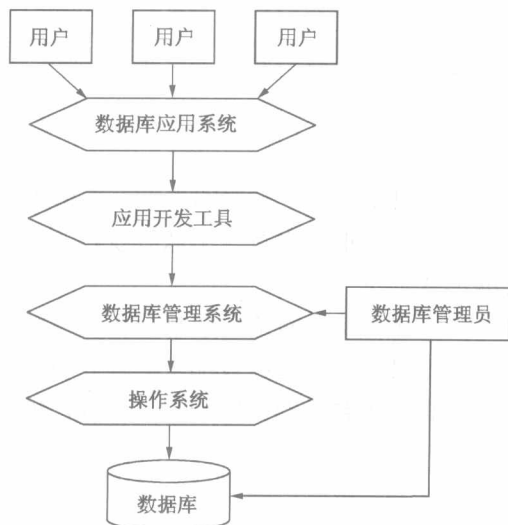


图 1.1 数据库系统的组成结构图

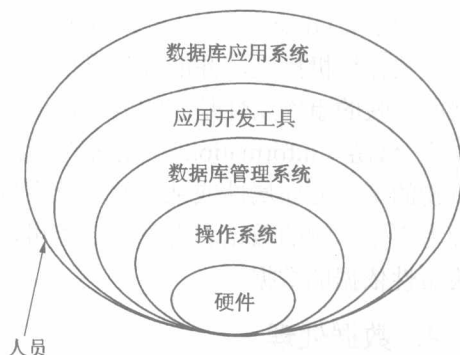


图 1.2 数据库管理系统在计算机中的地位

### 1.1.2 数据模型

数据模型是数据库系统的核心，也是定义数据库模型的基础，其好坏直接影响数据库的性能。

现实世界中的客观事物是彼此相互联系的。一方面，某一事物内部的诸因素和属性根据一定的组织原则相互具有联系，构成一个相对独立的系统；另一方面，某一事物同时也作为一个更大系统的一个因素或一种属性而存在，并与系统的其他因素或属性发生联系。客观事物的这种普遍联系性决定了作为事物属性记录符号的数据与数据之间也存在着一定的联系性。具有联系性的相关数据总是按照一定的组织关系排列，从而构成一定的结构，对这种结构的描述就是数据模型。

从理论上讲，数据模型是指反映客观事物及客观事物之间联系的数据组织的结构和形式。客观事物是千变万化的，各种客观事物的数据模型也是千差万别的，但也有其共同性。常用的数据模型有层次模型、网状模型和关系模型三种。

## 1. 层次模型

层次模型 (hierarchical model) 表示数据间的从属关系结构, 是一种以记录某一事物的类型为根节点的有向树结构。层次模型像一棵倒置的树, 根节点在上, 层次最高; 子节点在下, 逐层排列。这种用树形结构表示数据之间联系的模型也称为树结构。层次模型的特点是: 仅有一个无双亲的根节点; 根节点以外的子节点, 向上仅有一个父节点, 向下有若干子节点。

层次模型表示的是从根节点到子节点的一个节点对多个节点, 或从子节点到父节点的多个节点对一个节点的数据间的联系, 如图 1.3 所示。



图 1.3 层次模型

## 2. 网状模型

网状模型 (network model) 是层次模型的扩展, 表示多个从属关系的层次结构, 呈现一种交叉关系的网络结构。网状模型是以记录为节点的网络结构, 用网状数据结构表示实体与实体之间联系。网状模型的特点是: 可以有一个以上的节点无双亲, 至少有一个节点有多于一个的双亲。因此, 层次模型是网状模型的特殊形式, 网状模型可以表示较复杂的数据结构, 即可以表示数据间的纵向关系与横向关系。这种数据模型在概念上、结构上都比较复杂, 在操作上也有很多不便, 如图 1.4 所示。

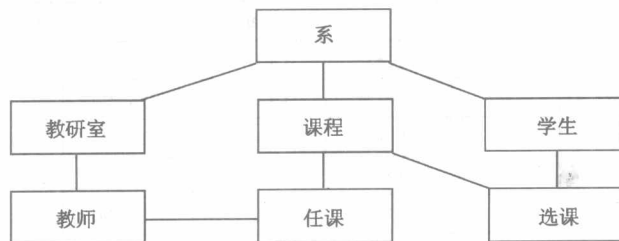


图 1.4 网状模型

## 3. 关系模型

关系模型 (relational model) 的所谓“关系”是有特定含义的。广义地说, 任何数据模型都描述一定事物数据之间的关系。关系中的每一个数据项也称“字段”是不可再分的, 它是最基本的单位; 并且每一竖列数据项的属性是相同的。列数根据需要而设, 并且各列的

顺序是任意的；每一横行记录由一个事物的诸多属性项构成。记录的顺序可以是任意的；一个关系是一张二维表，不允许有相同的字段名，也不允许有相同的记录行。

关系数据库采用人们经常使用的表格作为基本的数据结构，通过公共的关键字字段来实现不同二维表之间（或“关系”之间）的数据联系。可见关系模型呈二维表形式，如表1.1所示，这种结构简单明了，使用与学习起来都很方便（表中的“学号”、“姓名”等即为字段名）。

表 1.1 学生表

| 学号        | 姓名  | 专业编号 | 性别  | 出生日期       | 入学成绩 | 团员否 | 照片  | 简历  |
|-----------|-----|------|-----|------------|------|-----|-----|-----|
| 080301001 | 张跃林 | 03   | 男   | 1989-10-23 | 589  | T   | —   | —   |
| 080301020 | 张文斌 | 03   | 男   | 1989-5-20  | 593  | T   | —   | —   |
| 080302045 | 陈江城 | 03   | 男   | 1989-8-4   | 598  | T   | —   | —   |
| 084201002 | 夏利华 | 42   | 女   | 1989-9-4   | 497  | F   | —   | —   |
| 074202123 | 李林萍 | 42   | 女   | 1989-3-5   | 516  | T   | —   | —   |
| ...       | ... | ...  | ... | ...        | ...  | ... | ... | ... |

### 1.1.3 关系模型的基本概念

关系数据库（relation database）是若干个依照关系模型设计的数据表文件的集合。也就是说，关系数据库是由若干张完成关系模型设计的二维表组成的。一张二维表为一个数据表，数据表包含数据及数据间的关系。

一个关系数据库由若干个数据表组成，数据表又由若干个记录组成，而每一个记录是由若干个以字段属性加以分类的数据项组成的。图 1.5 中的教师表就是一个关系模型，它包含以下的概念。

| 教师编号 | 教师姓名 | 性别 | 职称  | 课程编号 | 工资      | 政府津贴 |
|------|------|----|-----|------|---------|------|
| JC01 | 陈得江  | 男  | 教授  | CC01 | 3310.00 | T    |
| JC02 | 赵惠兰  | 女  | 副教授 | CC01 | 2587.00 | F    |
| JC03 | 刘大林  | 男  | 讲师  | CC02 | 2000.00 | F    |
| JS01 | 秦健中  | 男  | 副教授 | CS01 | 3120.00 | T    |
| JS02 | 吴莲敏  | 女  | 讲师  | CS01 | 1500.00 | F    |

图 1.5 教师表

## 1. 关系

通常将一个没有重复行、重复列的二维表看成一个关系。一个关系就是一张二维表，每个关系有一个关系名。在 Visual FoxPro 中，一个关系即为一个表文件，其扩展名为 .dbf。

在 Visual FoxPro 中，一个关系对应于一个表文件，关系名则对应于表文件名或表名。

## 2. 元组

二维表的每一行在关系中称为元组。在 Visual FoxPro 中，一个元组对应表中的一个记录。

## 3. 属性

二维表的每一列在关系中称为属性，每个属性都有一个属性名，属性值则是各个元组属性的取值。在 Visual FoxPro 中，一个属性对应表中的一个字段，属性名对应字段名，属性值对应各个记录的字段值。

## 4. 域

属性的取值范围称为域。域作为属性值的集合，其类型与范围具体由属性的性质及其所表示的意义确定。如图 1.5 中“性别”属性的域是{男，女}。同一属性只能在相同域中取值。

## 5. 关键字

关系中能唯一区分、确定不同元组的属性或属性组合称为该关系的一个关键字。单个属性组成的关键字称为单关键字，多个属性组成的关键字称为组合关键字。需要强调的是，关键字的属性值不能取“空值”，因为无法唯一区分、确定元组。

## 6. 候选关键字

关系中能够成为关键字的属性或属性组合可能不是唯一的。凡在关系中能够唯一区分、确定不同元组的属性或属性组合都称为候选关键字。

## 7. 主关键字

在候选关键字中选定一个作为关键字，称为该关系的主关键字。在关系中，主关键字是唯一的。

## 8. 外部关键字

关系中某个属性或属性组合并非关键字，但却是另一个关系的主关键字，称此属性或属性组合为本关系的外部关键字。关系之间的联系是通过外部关键字实现的。外部关键字用来实现表与表之间的关联，实现表与表之间的参照完整性。

## 9. 关系模式

对关系的描述称为关系模式，其格式为

关系名(属性名 1, 属性名 2, ..., 属性名  $n$ )

关系既可以用二维表格描述，也可以用数学形式的关系模式来描述。一个关系模式对应一个关系的数据结构，也就是表的数据结构。

### 1.1.4 关系运算

关系的基本运算有两类：一类是传统的集合运算，包括并、差、交；另一类是专门的关系运算，包括选择、投影和连接。

#### 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结构模式，即结构相同。表 1.2 和表 1.3 给出了两种关系，用于介绍接下来的集合运算。

表 1.2 R 关系

| 学号        | 姓名  | 性别 |
|-----------|-----|----|
| 080301001 | 张跃林 | 男  |
| 080301020 | 张文斌 | 男  |
| 084201002 | 夏利华 | 女  |

表 1.3 S 关系

| 学号        | 姓名  | 性别 |
|-----------|-----|----|
| 084205255 | 李雅迪 | 女  |
| 080401081 | 孟江浩 | 男  |
| 084201002 | 夏利华 | 女  |

① 并：两个相同结构关系的并是由属于这两个关系的元组（记录）组成的集合，如表 1.4 所示。

② 差：有关系 R 和关系 S，是由属于 R 而不属于 S 的元组组成的集合，从 R 中去掉 S 中也含有的元组，如表 1.5 所示。

③ 交：有关系 R 和关系 S，是既属于 R 又属于 S 的元组组成的集合，如表 1.6 所示。

表 1.4 R ∪ S

| 学号        | 姓名  | 性别 |
|-----------|-----|----|
| 080301001 | 张跃林 | 男  |
| 080301020 | 张文斌 | 男  |
| 084201002 | 夏利华 | 女  |
| 084205255 | 李雅迪 | 女  |
| 080401081 | 孟江浩 | 男  |

表 1.5 R - S

| 学号        | 姓名  | 性别 |
|-----------|-----|----|
| 080301001 | 张跃林 | 男  |
| 080301020 | 张文斌 | 男  |

表 1.6 R ∩ S

| 学号        | 姓名  | 性别 |
|-----------|-----|----|
| 084201002 | 夏利华 | 女  |

#### 2. 专门的关系运算

在关系数据库中查询用户所需的数据时，需要对关系进行一定的关系运算。关系运算主要有选择、投影和连接三种。

① 选择：从关系找出满足条件的记录。是一种横向的操作，它可以根据用户的要求从关系中筛选出满足一定条件的记录，通过这种运算可以得到一个新的关系，但其中的元组是原关系的一个子集，不会影响关系的结构。例如 Visual FoxPro 中的“FOR<条件>”命令、“WHILE<条件>”命令等。

② 投影：从关系选取若干属性组成新的关系。是一种列的操作，它可以根据用户的要求从关系选出若干个字段组成新的关系，字段的个数或顺序往往不同。例如 Visual FoxPro 中的“FIELDS<字段 1, 字段 2, 字段 3, ...>”命令等。

③ 连接：将两个关系通过公共属性名连接成一个新的关系。连接运算可以实现两个关系的横向合并，在新的关系中可以反映出原来关系之间的联系。在连接运算中，按照字段值对应相等为条件进行的连接操作称为等值连接。自然连接是去掉重复属性的等值连接。

### 3. 关系数据库

关系数据库是由若干个依照关系模型设计的二维数据表文件的集合。一个关系数据库即为一个数据库文件。

### 4. 关系的完整性

关系的完整性是为保证数据库中数据的正确性和兼容性对关系模型提出的某种约束条件或规则。完整性通常包括实体完整性、参照完整性和域完整性，其中实体完整性和参照完整性是关系模型必须满足的完整性约束条件。

① 实体完整性：实体完整性是指关系的主关键字不能取“空值”。

② 参照完整性：参照完整性是定义建立关系之间联系的主关键字与外部关键字引用的约束条件。如“选课表”关系中课程号的值只能取“课程表”关系中已存在课程号的值。

③ 域完整性：实体完整性和参照完整性适用于任何关系型数据库系统，主要是对关系的主关键字和外部关键字取值必须做出有效的约束。域完整性则是根据应用环境的要求和实际需要，对某一具体应用所涉及的数据提出约束性条件。这一约束机制一般不应由应用程序提供，而应由关系模型提供定义并检验。如检验属性的类型、宽度等，进一步保证输入数据合理有效。域完整性主要包括字段有效性约束和记录有效性约束。

## 1.1.5 实体模型

### 1. 实体

客观存在并且可以相互区别的事物称为实体。实体可以是具体的，例如一个学生、一本书、一名教师；也可以是抽象的，例如一堂课、一次足球比赛。

### 2. 属性

描述实体的特征称为该实体的属性。例如，学生有学号、姓名、专业编号、性别、出生日期、入学成绩等方面的属性。属性有“型”和“值”之分，型即为属性名，值即为属性的具体内容（例如“084201002”、“夏利华”、“42”、“女”、“1989-9-4”、“497”、“F”）。

### 3. 实体间的联系

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，学生和课程是两个不同的实体，当学生选课时，两者之间则发生了关联，建立了联系——学生选择课程，课程被学生选学。实体间的联系分为两种：一种是实体内部各属性之间的联系，另一种是实体之间的联系。

### 1) 一对一联系 (1:1)

实体集 A 中的一个实体至多与实体集 B 中的一个实体相对应；反之，实体集 B 中的一个实体至多对应于实体集 A 中的一个实体，则称实体集 A 与实体集 B 为一对一联系。例如，厂长与工厂之间、观众与座位之间、旅客与车票之间、病人与病床之间等。

### 2) 一对多联系 (1:N)

实体集 A 中的一个实体与实体集 B 中的  $N$  ( $N \geq 0$ ) 个实体相对应；反之，实体集 B 中的一个实体至多与实体集 A 中的一个实体相对应。例如，学校与系、班级与学生、省与市等。

### 3) 多对多联系 (M:N)

实体集 A 中的一个实体与实体集 B 中的  $N$  ( $N \geq 0$ ) 个实体相对应；反之，实体集 B 中的一个实体与实体集 A 中的  $M$  ( $M \geq 0$ ) 个实体相对应。例如，教师与学生、学生与课程、工厂与产品、商店与顾客等。

## 练习与操作

练习 1 指出表 1.1 的“学生表”属于什么关系模型。

练习 2 指出表 1.1 的“学生表”中的“关系名”、“主关键字”、“属性”、“元组”、“域”。

练习 3 回答并完成下列问题：

- (1) 数据与信息的异同是什么？
- (2) 实体间联系的种类有哪些？
- (3) 层次模型、网状模型和关系模型的区别是什么？
- (4) 数据库能够把大量数据按一定的结构进行存储，\_\_\_\_\_，并实现数据共享。
- (5) 表由若干条记录组成，每一行称为一个\_\_\_\_\_，对应着一个真实对象的每一列称为一个“字段”。

## 1.2 Visual FoxPro 6.0 及其界面

### 1.2.1 Visual FoxPro 6.0 概述

Visual FoxPro 6.0 中文版作为一种关系型数据库系统，不仅简化了数据库的管理，在应用程序或数据库开发的任何领域都提供了帮助。而且它既支持编程方式又支持交互方式，使普通用户能方便地利用可视化的设计工具和向导来快速创建表、查询、视图、报表和表单等。Visual FoxPro 6.0 增强了以下的性能。

① 增强了项目和数据库管理能力。在 Visual FoxPro 6.0 中，对项目管理器和数据库设计器进行了改进，可以方便地管理项目和数据库。

② 改进了程序调试工具。在 Visual FoxPro 6.0 中，有一个更为简单的调试和修改程序，可以在主窗口打开调试工具，与应用程序界面一起显示；使用代码范围分析程序可以分析运行的程序代码。

③ 提供了更简单的表设计。在 Visual FoxPro 6.0 中，用户可以非常方便地利用表设计器来创建表，可以在创建表字段的同时方便地添加索引，指定多种默认值。通过表设计器中的选项卡，可以直接设定表的有效规则、触发器及统计值。