

普通高等院校本科计算机教材

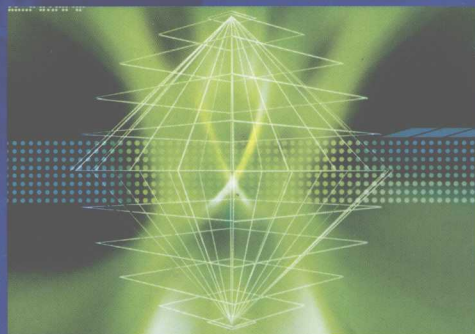
Visual FoxPro

程序设计教程

VISUAL FOXPRO

CHENGXU SHEJI JIAOCHENG

主 编 张 帆 张绪辉
副主编 王 芳 杜发启



华中科技大学出版社
<http://www.hustp.com>



TP311.138/544

2008

Visual FoxPro 程序设计教程

主 编 张 帆 张绪辉
副主编 王 芳 杜发启

主 编 张 帆 张绪辉

Visual FoxPro 程序设计教程

责任编辑：张帆
封面设计：王芳

责任编辑：张帆
封面设计：王芳

（武汉·华中）华中科技大学出版社发行

地址：武汉市武昌区洪山街 邮编：430074 电话：(027)87527437

本书由华中科技大学出版社出版
印刷：华中科技大学出版社印刷

定价：32.80元
000 000 张宇

本书由华中科技大学出版社出版
印刷：华中科技大学出版社印刷

华中科技大学出版社

中国·武汉

（本书由华中科技大学出版社出版，印刷：华中科技大学出版社印刷）

图书在版编目(CIP)数据

Visual FoxPro 程序设计教程/张 帆 张绪辉 主编. —武汉:华中科技大学出版社, 2008 年 2 月

ISBN 978-7-5609-4215-5

I. V… II. ①张… ②张… III. 关系数据库-数据库管理系统, Visual FoxPro-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(2008)第013458号

Visual FoxPro 程序设计教程

张 帆 张绪辉 主编

责任编辑:徐晓琦

封面设计:潘 群

责任校对:张 梁

责任监印:熊庆玉

出版发行:华中科技大学出版社(中国·武汉)

武昌喻家山 邮编:430074 电话:(027)87557437

录 排:华中科技大学惠友文印中心

印 刷:华中科技大学印刷厂

开本:787mm×1092mm 1/16

印张:19.5

字数:484 000

版次:2008 年 2 月第 1 版

印次:2008 年 2 月第 1 次印刷

定价:32.80 元

ISBN 978-7-5609-4215-5/TP·644

(本书若有印装质量问题,请向出版社发行部调换)

内 容 简 介

本书按照全国计算机等级考试二级考试(Visual FoxPro 程序设计)大纲的要求编写,内容全面,概念清晰,结构合理,对学习过程中可能遇到的疑难问题和易混淆概念作了详细的阐述。

本书主要包括 Visual FoxPro 基础、Visual FoxPro 操作基础、数据与数据运算、数据库及其操作、结构化查询语言 SQL、查询与视图、程序设计、表单设计、菜单设计、报表设计及开发应用等 11 个部分,并配有丰富的例题与习题。与本书配套的实践教材还有《Visual FoxPro 程序设计实训教程》。

本书可作为本、专科学生学习 Visual FoxPro 程序设计的教材,也可作为计算机等级考试教材。

前 言

Visual FoxPro 6.0 是一款优秀的小型数据库管理系统软件,它具有强大的数据库管理系统功能,完整而丰富的面向对象程序设计工具、较高的处理速度、友好的界面以及完备的兼容性,备受广大用户的欢迎。

本书根据 Visual FoxPro 程序设计的课程要求,结合计算机等级考试大纲,本着理论与实践并重的原则,由浅入深地介绍了关系数据库的基础知识,以及利用 Visual FoxPro 6.0 开发数据库应用系统的方法,并给出了大量的应用实例。全书共分 11 章,内容包括:Visual FoxPro 基础、Visual FoxPro 操作基础、数据与数据运算、数据库及其操作、关系数据库标准语言 SQL、查询与视图、程序设计基础、表单设计与应用、菜单设计、报表设计、开发应用程序等。

程序设计是一门实践性很强的课程,除了掌握基础的概念,编写程序和上机调试运行也是学习过程中的重要环节。与本书配套的《Visual FoxPro 程序设计实训教程》主要用于上机实训和等级实训。这两本教材相辅相成,如果能很好地掌握和消化这两本教材中的内容、完成习题,完全可以达到全国计算机二级等级考试(Visual FoxPro 程序设计)所要求的水平。

参加本书编写工作的有张帆、张绪辉、王芳、杜发启、焦启民、刘鸿翔等,本书中所给实例全部在 Visual FoxPro 6.0 环境下调试通过。

本书可作为本科及高职高专院校 Visual FoxPro 程序设计课程的教材,也可作为参加全国计算机二级等级考试(Visual FoxPro 程序设计)的教材。

由于编者水平有限,书中难免会有欠妥之处,恳请广大读者提出宝贵意见。

编 者

2008 年 1 月 20 日

目 录

第 1 章 Visual FoxPro 基础	(1)
1.1 数据库基础知识	(1)
1.2 关系数据库	(8)
1.3 数据库设计基础	(13)
1.4 Visual FoxPro 系统的发展	(16)
第 2 章 Visual FoxPro 6.0 操作基础	(21)
2.1 Visual FoxPro 6.0 的基本操作	(21)
2.2 项目管理器	(26)
2.3 Visual FoxPro 向导、设计器、生成器简介	(35)
第 3 章 数据与数据运算	(44)
3.1 Visual FoxPro 命令的工作方式及格式	(44)
3.2 数据类型	(45)
3.3 常量与变量	(45)
3.4 表达式	(52)
3.5 函数	(57)
第 4 章 数据库及其操作	(68)
4.1 Visual FoxPro 数据库及其建立	(68)
4.2 建立表	(73)
4.3 表的基本操作	(79)
4.4 索引	(85)
4.5 数据完整性	(89)
4.6 自由表	(90)
4.7 多表的操作	(92)
4.8 排序	(96)
第 5 章 关系数据库标准语言 SQL	(99)
5.1 SQL 概述	(99)
5.2 SQL 与数据库模式结构的关系	(100)
5.3 查询功能	(101)
5.4 操作功能	(126)
5.5 定义功能	(128)

第 6 章 查询与视图	(137)
6.1 查询	(137)
6.2 视图	(148)
第 7 章 程序设计基础	(158)
7.1 程序与程序文件	(158)
7.2 程序的基本结构	(166)
7.3 多模块程序	(178)
7.4 程序调试	(185)
第 8 章 表单设计与应用	(194)
8.1 表单设计	(194)
8.2 面向对象的概念	(203)
8.3 常用表单控件	(212)
8.4 一个表单应用的实例	(238)
第 9 章 菜单设计	(244)
9.1 Visual FoxPro 系统菜单	(244)
9.2 下拉式菜单设计	(247)
9.3 快捷菜单设计	(258)
9.4 用编程的方式定义菜单	(260)
第 10 章 报表设计	(265)
10.1 创建报表	(265)
10.2 设计报表	(269)
第 11 章 开发应用程序	(285)
11.1 应用项目综合实践	(285)
11.2 应用程序的帮助文件	(293)
11.3 发布应用程序	(296)

第 1 章

Visual FoxPro 基础

Visual FoxPro 6.0(简称 VFP 6.0)是 Xbase 数据库家族的一员,可在 Windows 95 及以上版本操作系统中运行。它采用了可视化、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,提高了系统的模块性和紧凑性,是目前被广泛应用的 32 位关系型数据库管理系统。

Visual FoxPro 6.0 可以对大量数据进行存储、查找、统计、增加、删除、修改等操作,还可以制作报表和标签,用户甚至可以不用编写复杂的程序便完成应用程序的开发工作。

本章将介绍数据库的基本概念、关系数据库设计的基础知识及 Visual FoxPro 的特点等,为读者学习后续知识打下基础。

1.1 数据库基础知识

1.1.1 数据与数据处理

数据(Data):是指存储在某一媒体上能够识别的物理符号。它是对客观事物某些特征及相互联系的一种抽象化、符号化的表示。

通常见到的 EXCEL 表格就是数据的代表之一,如一张人事档案登记表,可以包括某人的姓名、身高、出生日期、职称等。所以数据的概念包括两个方面:其一是数据内容,如上面所述 EXCEL 表中的具体内容,这些数据内容能描述事物的特性;其二是存储在某一媒体上的数据形式。

同一个数据可以有多种表示方法,例如,某人的出生日期是“1980 年 11 月 1 日”,也可以表示为“80/11/1”或“80-11-1”等,表示方法虽然不同,但其含义是一样的。

数据既可以是包括数字、字母、文字和其他特殊字符的文本形式的数据,也可以是包括图形、图像、动画、影像、声音等的多媒体数据,但是使用最多、最基本的仍然是文本数据。

数据处理:是指将数据转换成信息的过程。现实生活中的数据往往是原始的、非规范化的,通过对这些原始数据进行处理加工后所产生新的数据通常称为信息。数据处理的过程包括数据的收集、记录、分类、排序、存储、加工等,处理后的数据是一种精炼的数据,能够反映事物现象的本质特征及内在联系。

由此可知,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。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,一个人的出生日期属于原始数据,而年龄是通过当前年份与出生日期之

间经过简单的计算而得到的二次数据,根据年龄、性别、职称等有关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、光盘等)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。

1.1.2 计算机数据管理

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。计算机数据管理随着计算机件、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

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各负其责。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,计算机广泛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。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专门管理外存储器的数据。

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,而不必关心数据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性。

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

20 世纪中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。文件系统在数据管理方面已无法适应开发系统的需要。为了实现计算机对数据的统一管理,达到数据共享的目的,发展了数据库技术。

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。

为数据库的建立、使用和维护而配置的软件称为数据库管理系统 DBMS(Database Management System)。数据库管理系统利用了操作系统提供的输入/输出控制和文件访问

功能,因此它需要在操作系统的支持下运行。Visual FoxPro 就是一种在微机上运行的数据库管理系统软件。

面向(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。网络技术的进展为数据库提供了分布式运行环境,从主机—终端系统结构发展到客户/服务器(client/server)系统结构。

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 DBMS 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节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节点上,使大部分数据可以就地或就近存取,而用户感觉不到数据的远程分布。

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,各个节点上的计算机可以利用网络通信功能访问其他节点上的数据库资源。它一般由两部分组成:一是本地节点的数据,二是本地节点共享的其他节点上有关的数据。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。节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。

Visual FoxPro 为创建功能强大的客户/服务器应用程序提供了一些专用工具。客户/服务器应用程序具有本地(客户)用户界面,但访问的是远程服务器上的数据。此应用程序根据前端和后端产品的能力将工作分布到本地机和服务器,可以将 Visual FoxPro 功能强、速度快、图形化的用户界面以及高级的查询、报表和处理等优点与 ODBC 数据源或服务器的本地语法等功能紧密地结合在一起。Visual FoxPro 服务器之间的协作可以为用户提供功能强大的客户/服务器解决方案。

开放式数据库连接(Open Database Connectivity, ODBC)是用于数据库服务器的一种标准协议。可以安装多种数据库的 ODBC 驱动程序,从而使 Visual FoxPro 能够与该数据库相连,访问库中的数据。如果选择“完全安装”或“用户自定义安装”安装选项,则可以获得“开放式数据库连接”支持。使用 ODBC,可以从 Visual FoxPro 中访问 SQL Server 数据源。但是,必须先定义数据源才能进行访问。

类—(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 8 章中详细介绍面向对象的基本概念。

1.1.3 数据库系统

1. 数据库的概念

数据库(Database)是存储在计算机存储设备上并可共享的相关数据的集合。可将其通俗的理解为存放数据的仓库,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。

数据库中的数据往往不像文件系统那样,只面向某一项特定应用,它们面向多种应用,可以被多个用户、多个应用程序共享,其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。

在 Visual FoxPro 中,数据库文件的扩展名为 .dbc,而数据库表是一个扩展名为 .dbf 的文件,将一些有关联的表按一定方式组织在一起,便得到一个数据库文件,也就是说,在 Visual FoxPro 中数据库是以表为基础的,是一些有关联的表的集合。

2. 数据库系统

数据库系统(Database System, DBS)是指带有数据库的计算机系统,即引进数据库技术后实现有组织地、动态地存储大量相关数据,并提供数据处理和信息资源共享的计算机系统。

一般来说,数据库系统由下面几个部分组成。

① 计算机硬件系统:是指能运行操作系统、数据库管理系统、应用程序等,并且具有数据存储能力的本地计算机系统或网络硬件环境。

② 数据库集合:是指存储在本地计算机外存设备或网络存储设备上的若干个设计合理、满足应用需要的数据库。

③ 数据库管理系统(Database Management System, DBMS):是数据库系统的核心,是用户用来建立、使用和管理数据库的软件系统,它属于计算机系统软件,具有数据库定义、操作、运行和维护等功能。DBMS 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,使得数据与程序具有较高的独立性,方便用户以交互命令或程序方式对数据库进行操作、对大量的数据进行管理和存储。

④ 数据库应用系统(Database Application System, DBAS):是指用户为了解决某一类信息处理的实际问题而利用数据库系统开发的软件系统,一个 DBAS 通常由数据库和应用程序两部分组成,它们都需要在 DBMS 支持下进行开发。数据库应用系统往往面向某类实际应用,例如,以数据库为基础的财务管理系统、人事管理系统、图书管理系统、教学管理系统、生产管理系统,等等。

3. 数据库系统的特点

在数据库技术出现之前,计算机使用文件系统来存储、管理数据。与文件系统比较,数据库系统的主要特点如下。

(1) 数据结构化

在文件系统中,虽然对于单个文件其数据是有结构的,但从整个系统来看,各文件的数

据相互之间却没什么联系。数据库系统则不同,不但在同一数据库中的数据文件彼此关联,且在整体上服从一定的结构形式。

所以数据库系统不仅可以表示事物内部各数据项之间的联系,也可以表示事物与事物之间的联系,从而反映出现实世界事物之间的本质关系。

(2) 数据共享

在文件系统中,数据通常是由特定的用户专用,而在数据库系统中的数据,最重要的特点之一就是能实现数据的共享,并通过数据库管理系统来统一管理。数据库中的数据可以为同一单位的各部门之间所共享,甚至也可为不同单位、地域的用户所共享。

建立数据库时,应当以面向全局的观点组织数据库中的数据或数据文件,使各数据文件中的数据能在数据库系统中实现资源共享。

(3) 数据独立性

在文件系统中,数据结构和应用程序之间相互依赖,一方的改变往往会影响到另一方,数据库系统则力求减少这种依赖。数据库管理系统提供的映象功能令应用程序与数据的逻辑结构、物理结构间具有较高的独立性。

(4) 可控冗余度

当数据专用时,每个用户拥有并使用自己的数据,这样就不可避免地会出现许多数据重复,即数据冗余现象。由于数据库可以被多个用户或应用程序共享,数据的存取往往是并发的,即多个用户同时使用同一个数据库,所以不必要的重复数据将会消除。但有时为了提高查询效率,往往会保留少量的重复数据,其冗余度可由设计人员控制。

1.1.4 数据模型

对数据及其相互之间关系的描述称为数据模型。数据模型的构造方法决定了数据库中数据之间的关联方式,决定了数据库的设计方法。

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、概念化到数据库里的具体表示的逐级抽象过程。

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;在体育竞赛中有参赛代表队、竞赛项目,代表队中的运动员参加特定项目的比赛,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。

客观存在并且可以相互区别的事物称为实体。实体可以是具体的人、实际的事物,也可能是抽象的事件。比如,一个职工、一个部门等属于实际事物,而订货、借阅图书、授课等是比较抽象的事件。

描述实体的特性称为属性,属性是实体之间相互区别的标志,一个实体可以由若干个属性来描述,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、单价等多个属性来描述。

相同类型实体的集合称为实体集。例如,在职工实体集当中,(09861, 吴大伟, 男,

55/12/06,教授)是教工名册中的一个具体教师,而所有教师的集合就是一个实体集。

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 中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,也就是说,每一条记录表示一个实体。

2. 实体联系

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。两个实体间的联系可以归结为以下三种类型。

(1) 一对一联系(one to one relationship)

对于实体 A 中的任一个元素,实体 B 中至多有一个实体与之联系,反之亦然。

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,一个单位人事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,如图 1-1 所示。

(2) 一对多联系(one to many relationship)

对于实体 A 中的任一个元素,实体 B 中可以有 $n(n \geq 1)$ 个实体与之联系;反之,对于实体 B 中的任一个元素,实体 A 中至多只有一个实体与之对应。例如,考查部门和职工两个实体集,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,如图 1-2 所示;考查学生和院系两个实体集,一个学生只能在一个院系里注册,而一个院系有很多个学生,院系和学生也是一对多的联系。

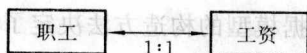


图 1-1 一对一联系

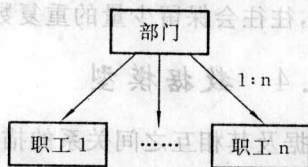


图 1-2 一对多联系

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。一对多联系是最普遍的联系,也可以把一对一的联系看作一对多联系的一个特殊情况。

(3) 多对多联系(many to many relationship)

对于实体 A 中的任一个元素,实体 B 中可以有 $n(n \geq 1)$ 个实体与之联系;反之,对于实体 B 中的任一个元素,实体 A 中也可以有 $m(m \geq 1)$ 个实体与之对应,这种情况称为多对多联系。

例如,一个成绩数据库中,有学生和课程两个实体集,一个学生可以选修多门课程,而一门课程由多个学生选修,因此,学生和课程之间就是多对多的联系,如图 1-3 所示。

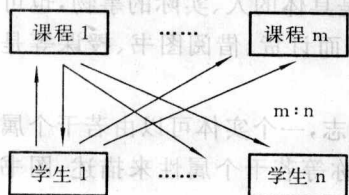


图 1-3 多对多联系

在 Visual FoxPro 中,多对多的联系表现为一个表中的多条记录在相关表中同样有多条记录与其关联,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。例如,在一张订单中可以包括多项商品,因此对于订单表中的每条记录,在商品表中

可以有多条记录与之对应;同样,每项商品也可以出现在多个订单中,因此对于商品表中的每条记录,在订单表中也有多条记录与之对应。

3. 数据模型

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。在数据库用数据模型来抽象、处理和表示现实世界中的数据和信息。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。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,不同的数据模型实际上是提供模型化数据和信息的不同工具。数据库管理系统所支持的数据模型可分为三种:层次模型、网状模型、关系模型。

(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,数据被组织成由“根”开始的“树”,如果不再向下分支,那么此分支序列中最后的结点称为“叶”,每个实体由根开始沿着不同的分支放在不同的层次上,实体类型之间只有简单的一对多的层次联系,即树根与树枝的关系,或称父子关系,而同一双亲的子女结点称为兄弟结点。如图 1-4 所示,A 为根结点,B、D 为兄弟结点,是 A 的子女结点,E、G 也为兄弟结点,是 D 的子女结点,C 与 F 都是叶子结点。

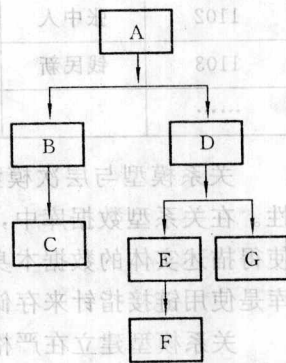


图 1-4 层次模型

这种模型的优点是简单、直观、处理方便,适用于表现具有比较规范的层次关系的结构,缺点是不能直接表现含有多对多关系的复杂结构。

支持层次数据模型的 DBMS 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。

(2) 网状模型

用网状结构表示实体间联系的模型称为网状模型,网状结构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,它允许结点有多于一个的父结点,并且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记录类型之间多对多的联系,但其结构比较复杂,数据处理比较困难。

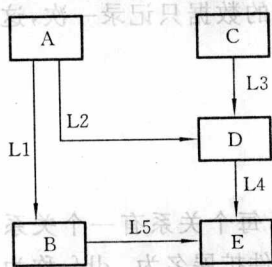


图 1-5 网状模型

图 1-5 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。由于在网状模型中,子女结点与双亲结点的联系可以不唯一,因而要为每个联系命名,并指出与该联系有关的双亲记录和子女记录。如图 1-5 中的 L1~L5 都是相应关系的联系命名。

网状模型的主要优点是表示多对多的联系时具有很大的灵活性,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 DBMS 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同样的。从逻辑上

看,它们都是用结点表示实体,用有向箭头表示实体间的联系;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) 关系模型

在关系模型中,数据的逻辑结构实际上是一张二维表,这种二维表由行与列组成,如表 1-1 所示。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,每一张二维表就是一个关系,它不仅能描述实体本身,而且还能反映实体之间的联系。

表 1-1 人事档案表

职工号	姓名	性别	出生日期	是否党员		备注
1101	周萍	女	01-12-69	. F.		
1102	张中人	男	11-05-77	. T.		
1103	钱民新	男	01-07-67	. T.		
.....						

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、模型概念的单一性。在关系型数据库中,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。

关系模型建立在严格的数学理论基础之上,其数据结构简单、清晰,易于操作和管理。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了最广泛的应用。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 就是这样的一种关系数据库管理系统。

1.2 关系数据库

关系数据库就是用关系模型构成的数据库,这种数据库由一个或多个数据表组成,各数据表之间可以建立相互联系。图 1-6 所示为用 Visual FoxPro 创建的一个教务管理数据库,该数据库由三个(或更多)的表组成,各表之间通过公共属性联系起来,如学生表与成绩表可以通过公共属性“学号”字段建立联系。

在关系数据库中,数据被分散到不同的数据表中,使每一个表中的数据只记录一次,这样便可避免数据的重复输入,减少了数据冗余。

1.2.1 关系术语

1. 关系

一个关系通常对应于我们所熟悉的一张二维表,如表 1-1 所示,每个关系有一个关系名,如人事档案。在 Visual FoxPro 中,一个关系存储为一个文件,文件扩展名为 .dbf,称为“表”。

关系可用关系模式来描述,一个关系模式对应一个关系的结构,其格式为:

The figure displays three overlapping Visual FoxPro database window screenshots. The top-left window shows the '学生' (Students) table with columns: 学号 (Student ID), 姓名 (Name), 性别 (Gender), 出生日期 (Date of Birth), 民族 (Nationality), and 政治面貌 (Political Status). The top-right window shows the '成绩' (Grades) table with columns: 学号 (Student ID), 成绩 (Grade), 课程代号 (Course Code), 开课学期 (Semester), and 考试类别 (Exam Type). The bottom window shows the '课程' (Courses) table with columns: 课程名称 (Course Name), 学分 (Credits), 课程代号 (Course Code), and 学时 (Hours).

学号	姓名	性别	出生日期	民族	政治面貌
2004010001	叶仔	女	01/02/80	汉族	团员
2004010002	赵军	男	01/04/81	回族	团员
2004020001	王丽	女	12/27/80	汉族	团员
2004020002					
2004020003					
2004030001					
2004040001					

学号	成绩	课程代号	开课学期	考试类别
2004010001	87.5	0001	1	考试
2004010001	67.0	0002		考察
2004010002	89.			
2004020003	90.			
2004020003	45.			
2004020003	56.			
2004040001	98.			

课程名称	学分	课程代号	学时
英语	5.0	0001	92
计算机基础	1.5	0002	54
高等数学	3.0	0003	92
数据结构	2.0	0004	72
C语言程序设计	1.5	0005	72
操作系统	2.0	0006	72
计算机网络	2.0	0007	72

图 1-6 教务管理数据库中的表

关系名(属性名 1,属性名 2,...,属性名 n)

在 Visual FoxPro 中表结构表示为:

表名(字段名 1,字段名 2,...,字段名)

如表 1-1 的关系表结构可描述为:

人事档案(职工号,姓名,性别,出生日期,是否党员,...,备注)

在关系模型中,实体以及实体间的联系都是用关系来表示的。

2. 元组

在一个二维表所表示的具体关系中,水平方向的行称为元组,即每一行是一个元组。元组对应表文件中的一个具体记录。例如,如表 1-1 所示的人事档案表中有多少行,就包括了多少条记录,也就有多少个元组。

3. 属性

二维表中垂直方向的列称为属性,每一列有一个属性名,如职工号、姓名、性别等。在 Visual FoxPro 中表示为字段名。每个字段都有三个重要的要素:字段名、数据类型和宽度,它们在创建表结构的时候进行定义。例如,人事档案表的部门号、姓名、性别等都是字段名,它们和相应的数据类型及宽度组成表的结构。

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称为域。例如,“姓名”的域是文字字符,“性别”的域是(男,女),“是否党员”可视为逻辑型属性,其域只能是逻辑真或逻辑假。

5. 主关键字

主关键字是属性或属性的组合,在 Visual FoxPro 中表示为字段或字段的组合,它唯一标识一个元组。如人事档案表中的职工号可以作为唯一标识一条记录的关键字,而性别字段就不能作为起唯一标识作用的关键字。在 Visual FoxPro 中,主关键字和候选关键字都起唯一标识一个元组的作用。

6. 外部关键字

外部关键字是用来与一个关系进行连接的字段。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(或属性)就称为外部关键字。如图 1-6 中成绩表中的学号就可作外部关键字,而学生表中的学号则是主关键字。

从集合论的观点来定义关系,可以将关系定义为元组的集合,关系模式是命名的属性集合,元组是属性值的集合,一个具体的关系模型是若干个有联系的关系模式的集合。

在 Visual FoxPro 中,把相互之间存在联系的表放到一个类型为 .dbc 的数据库中统一管理。例如,在职工管理数据库中可以加入职工表、工资表,在图书管理数据库中可以加入读者表、图书表、借阅表。

1.2.2 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 关系必须规范化。所谓规范化是指关系模型中的每一个关系模式都必须满足一定的要求。最基本的要求是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。

手工制表中经常出现如表 1-2 所示的复合表,这种表格不是二维表,不能直接作为关系来存放,只要去掉表 1-2 中的应发工资和应扣工资两个表项就可以了。如果有必要,在数据输出时可以对打印格式另行设计,从而满足用户的要求。

表 1-2 复合表

职工号	姓名	日期	应发工资				应扣工资					实发工资
			岗位	薪级	补贴	奖金	房租	水费	电费	公积金	所得税	

② 在同一个关系中不能出现相同的属性名,Visual FoxPro 不允许同一个表中有相同的字段名。

③ 关系中不允许有完全相同的元组,即冗余。

④ 在一个关系中元组的次序无关紧要。也就是说,任意交换两行的位置并不影响数据的实际含义。日常生活中经常见到的“排名不分先后”正反映这种意义。

⑤ 在一个关系中列的次序无关紧要。任意交换两列的位置也不影响数据的实际含义。例如,工资单里奖金和基本工资哪一项在前面都不重要,重要的是实际数额。

1.2.3 实际关系模型

一个具体的关系模型由若干个关系模式组成。在 Visual FoxPro 中,一个数据库中包含相互之间存在联系的多个表,它代表一个实际的关系模型。为了反映出各个表所表示的实体之间的联系,公共字段名往往起着“桥梁”的作用。这仅仅是从形式上看,实际分析时,应当从语义上来确定联系。

例 1-1 部门—职工—工资关系模型。

设职工管理数据库中有以下三个表:

关注部门(部门编码,部门名称,负责人,地址,电话,邮政编码)