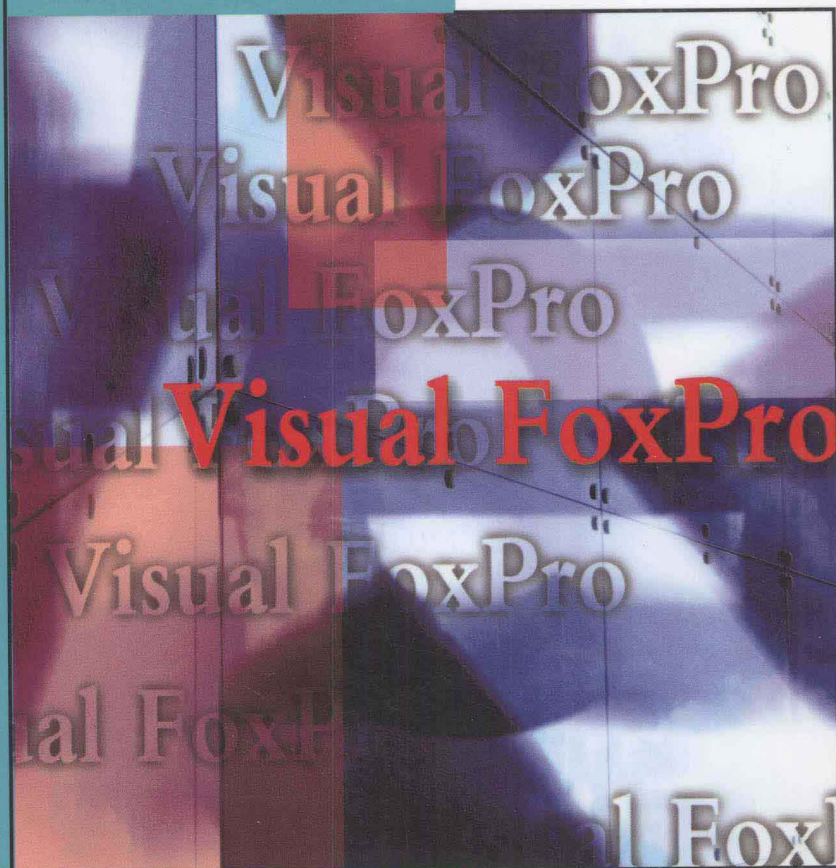


21世纪应用型人才培养系列教材

Visual FoxPro JI CHU JIAO CHENG

冯端品 主编



Visual FoxPro 基础教程



高等教育出版社
HIGHER EDUCATION PRESS

21 世纪应用型人才培养系列教材

Visual FoxPro 基础教程

冯端品 主编

高等教育出版社

内容提要

本书是应用型人才培养系列教材之一,是面向 21 世纪课程教材。该系列教材根据应用型人才培养的教学基本要求,并参照有关行业最新颁发的职业鉴定规范及高级工等级标准编写。本书主要介绍了数据库基本概念,VFP 基础知识、基本操作,VFP 程序设计和应用实例的编写方法等内容,附录部分辑录了常用的 VFP 命令与函数。本书适合作为高等职业学校、部分本科院校的计算机及相关专业教学用书,也可作为中高级职业资格与就业培训用书。

图书在版编目(CIP)数据

Visual FoxPro 基础教程 / 冯端品主编. —北京: 高等教育出版社, 2003.7
ISBN 7-04-012055-0

I. V… II. 冯… III. 关系数据库—数据库管理系统, Visual FoxPro—高等学校: 技术学校—教材
IV. TP311.138

中国版本图书馆 CIP 数据核字(2003)第 047624 号

责任编辑 黄红英 司马镭

封面设计 吴 昊

责任印制 潘文瑞

书 名 Visual FoxPro 基础教程
主 编 冯端品

出版发行 高等教育出版社
社 址 北京市西城区德外大街 4 号
邮政编码 100011
总 机 010-82028899
传 真 021-56965341

购书热线 010-64054588
021-56964871
免费咨询 800-810-0598
网 址 <http://www.hep.edu.cn>
<http://www.hep.com.cn>
<http://www.hepsh.com>

排 版 南京理工排版校对公司
印 刷 上海三印时报印刷有限公司

开 本 787 × 1092 1/16
印 张 16.25
字 数 386 000

版 次 2003 年 7 月第 1 版
印 次 2005 年 2 月第 3 次
定 价 20.50 元

凡购买高等教育出版社图书,如有缺页、倒页、脱页等质量问题,请在所购图书销售部门联系调换。

版权所有 侵权必究

前 言

近年来,随着我国教育的不断深入,对计算机教材提出了新的要求。特别是高等职业学校和高等专科学校,原来差不多都是使用大学本科的教材,很少有适合自己特点的计算机教材。因此,教师们热切盼望能有专为高等职业学校、高等专科学校使用的教材出版。

正是在这一背景下,我们组织具有多年教学经验的老师,参照 2002 年公布的国家计算机等级考试二级 Visual FoxPro 考试大纲,编写了这本主要为高等职业学校和高等专科学校使用的教材。本书也可供其他院校和自学 Visual FoxPro(简称 VFP)的读者使用。

本书力图把 Visual FoxPro 的最基本的知识和使用方法介绍给读者。针对高等职业学校和高等专科学校学生的特点,既有数据库的相关理论,又有数据库的实际操作和程序设计。编写中淡化理论,重在实际应用。程序设计则重在编程的基本思路和基本方法,不对学生作过高的要求。

本书结构新颖,层次分明,内容由浅入深。重要的概念和知识点都有实际例子,便于读者理解。重要的操作都有菜单和命令两种操作方式,供不同的读者使用。在程序设计中,摒弃了把控件使用与程序设计分开讲授的做法,把控件的使用分散到程序设计中,先介绍最常用的控件,随着程序设计的深入,逐步介绍其他控件。

第四章多表操作和第七章程序设计进阶属于学习 Visual FoxPro 6.0 较高的要求,在编排时,特地把这些知识相对集中,教师们可根据各专业的特点和课时数的多少灵活处理,选学或不学该部分内容。

各章末有小结,归纳该章的重要概念和重要知识点,供读者复习使用。章末还附有丰富的习题和上机操作题,供读者课后练习和上机操作使用。书末的附录中列出了 Visual FoxPro 的常用命令和常用函数,供读者查阅。

本书的主编为冯端品,副主编为国雪飞、刘睿;主审为任泰明。参加本书编写的还有:卓晓波、张殿伟、巫鸿宾、罗亚东、杨闯、明文、尹华国、余辉。

由于时间紧迫,作者水平有限,缺点和错误在所难免,欢迎读者和专家指正。

作 者
2003 年 4 月

目 录

1	第 1 章 Visual FoxPro 基础知识
1	1.1 数据库基本概念
7	1.2 关系数据库
9	1.3 Visual FoxPro 系统特点与工作方式
16	1.4 Visual FoxPro 的基本数据元素
25	习题
30	第 2 章 表和数据库的基本操作
30	2.1 数据库、自由表和数据库表
32	2.2 创建自由表
39	2.3 创建数据库表
50	2.4 打开、修改和关闭数据库
51	本章小结
52	习题
59	第 3 章 表记录的操作
59	3.1 表的打开和关闭
61	3.2 追加记录
62	3.3 浏览记录和记录指针
68	3.4 修改记录
69	3.5 删除记录
70	3.6 恢复记录
71	3.7 索引
80	3.8 记录的查找
81	3.9 记录的统计
83	本章小结
84	习题
91	第 4 章 多表操作
91	4.1 工作区的选择和使用
94	4.2 表的关联
101	4.3 参照完整性
106	4.4 表的连接

107	4.5 表间的数据更新
108	4.6 建立表间临时关联
112	本章小结
113	习题
119	第5章 查询、视图和报表
119	5.1 查询
126	5.2 视图
130	5.3 关系数据库标准语言 SQL
134	5.4 报表
142	本章小结
144	习题
152	第6章 Visual FoxPro 程序设计
152	6.1 程序设计基础
162	6.2 顺序程序设计
167	6.3 分支程序设计
174	6.4 循环程序设计
183	本章小结
185	习题
192	第7章 Visual FoxPro 程序设计进阶
192	7.1 过程和函数程序设计
200	7.2 数据环境与数据绑定
202	7.3 表格、页框和容器控件
209	7.4 表单集与多重表单
214	7.5 项目管理器
216	7.6 创建菜单
221	7.7 综合程序设计方法简介
223	本章小结
224	习题
227	附录1 Visual FoxPro 常用命令
245	附录2 Visual FoxPro 常用函数

第 1 章

Visual FoxPro 基础知识

Visual FoxPro 6.0(中文版)是由 Microsoft 公司于 1998 年推出的产品,它是一种可运行于 Windows95/98、Windows NT 平台的 32 位数据库管理系统,主要用于数据库的管理和数据库应用程序的开发。无论是组织信息、运行查询、创建关系型数据库系统,还是为最终用户编写功能全面的数据管理应用程序,Visual FoxPro 6.0 都可以提供管理数据所需的工具,可以在应用程序或数据库开发的各个领域提供帮助。Visual FoxPro 6.0 所具有的速度、能力和灵活性,是普通数据库管理系统无法比拟的。

本章主要介绍学习和使用 Visual FoxPro 6.0 必不可少的最重要的基础知识。根据大纲的要求,本章要求学生掌握的知识有:(1)数据、数据库、数据库管理系统、数据库系统、数据模型、面向对象数据库等基本概念;(2)关系数据库的有关知识,包括关系、元组、属性、域、关系模式、关键字等,关系运算包括并、差、交、选择、投影、联接以及关系的完整性;(3)Visual FoxPro 6.0 的系统特点与工作模式,主要包括 Visual FoxPro 6.0 的启动、退出,向导、设计器和生成器等可视化的管理工具及两种工作方式;(4)Visual FoxPro 6.0 的基本数组元素,主要包括常量、变量、函数和表达式等。

1.1 数据库基本概念

信息在现代社会和经济发展中起着越来越重要的作用,信息资源的开发和利用水平已成为衡量一个国家综合国力的重要标志。在当今的社会里,由于信息的急剧增加,并且信息的形式多种多样、结构复杂,使信息处理已成为管理工作中一个极其重要的问题。在计算机应用领域中,数据处理是一个重要方面,约占 70% 以上。数据库技术就是作为数据处理中的一门技术而迅速发展起来的。

数据处理是指对各种类型的数据进行收集、存储、分类、计算、加工、检索和传输的一系列过程。数据处理的目的是对“原始数据”进行加工,从中得出对人们有价值、有意义的数据,以此作为行动和决策的依据,同时也可对这些数据进行编辑,并储存起来,以便今后能够充分地利用这些资源。

数据库技术实现了对所有数据实行统一的、集中的、独立的管理,即科学地组织和存储数据,从而可以高效地获取数据。数据库技术作为计算机科学和技术中的一个重要分支,它和计算机网络、人工智能一起称为当今计算机的三大技术,成为各种计算机应用系统的核心部分。

1.1.1 数据、数据库、数据库管理系统和数据库系统

1. 数据、信息

在日常生活中,所有的数学数值被称为数据,但计算机处理的数据和数学中的数据有很

大的区别。在计算机处理的数据中,数据是指存储在某一种媒体上能被识别的符号。数据的概念包括两个方面:一是描述事物特性的数据内容;二是存储在某一种媒体上的数据形式。由于描述事物特性必须借助一定的符号,这些符号就是数据形式。数据形式可以是多种多样的,例如某人的出生日期是“1985年8月15日”,可以将该形式写成为“08/15/85”来表示,其含义并没有变化。

数据既包括价格、工资、产量之类的数值型数据,又包括单位、姓名、句子、文章等非数值的字符型数据。数据的类型很多,具体内容会在第二章中详细介绍。

数据经过提炼和抽象之后具有使用价值才能成为信息,并仍然以数据的形式出现。因而数据是信息的具体表现形式,是信息的载体,是组成信息的单元,是人们认识信息的一种媒介。例如表 1-1 列出了某班学生学习成绩表,它反映了学生学习情况这一信息。

表 1-1 学生成绩表

学号	姓 名	数学	英语	VFP 程序设计	电子技术	体育
2314	黄 妮	87	67	78	77	90
0143	王为国	67	78	87	69	88
0123	马文广	56	78	89	98	89
1243	王 伟	67	89	54	78	99
0032	张丽美	79	65	78	98	67
1023	李佳丽	98	87	81	91	94
0065	孙小伟	62	34	65	72	65

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。可以用下面的式子简单地表示出信息与数据之间的关系:

信息=数据+处理

信息和数据是密切相关、不可分离的,但它们也有区别。一是信息是依赖于数据而存在的,并比数据更直接、更基本地反映现实;二是并非任何数据都能够表示信息,只有在数据条理化后,才成为信息的具体表现形式。

2. 数据库

数据库(database)是存储在计算机存储设备上,结构化、相关数据的集合,也可以看成是长期存储在计算机内、有组织的、可共享的数据集合。数据库不仅包括描述事物的数据本身,也包括相关事物之间的关系。

数据库就相当于一个存放数据的大仓库,这个大仓库中存放着具有一定关系的数据。在计算机应用中,这个大仓库是以文件的形式存放在计算机的存储介质上的。如表 1-1 就可以看成是一个数据库表。

3. 数据库管理系统

数据库管理系统(database management system,简称 DBMS)是为数据库的建立、使用、管理和维护而配置的软件,是介于用户与操作系统之间的完成数据管理的系统软件。一般来说,它的主要功能包括以下几个方面:

(1) 数据定义功能

提供“数据定义语言”(DDL),用户通过它可以方便地对数据库中的相关内容进行定义。例如,对数据库、表、索引进行定义。

(2) 数据操作功能

提供“数据操作语言”(DML),支持用户对数据库中的数据进行添加、删除、修改、查询等操作。

(3) 数据库运行控制功能

包括并发控制、安全性检查、完整性约束条件的检查和执行、数据库恢复以及数据库的内部维护,这是 DBMS 的核心部分。

(4) 数据库的建立和维护功能

主要包括数据库初始数据的输入、转换功能,数据库的存储、恢复功能,数据库的重新组织功能和性能监视、分析功能等。

应当指出,DBMS 不是独立进行工作的,它在操作系统支持下才能进行工作,它的安装也是在操作系统安装后才能进行。在工作过程中,可以很方便地由 DBMS 返回到操作系统,而且任何一种 DBMS 都与确定的操作系统相配合,缺少操作系统则 DBMS 就无法工作。例如 Visual FoxPro 6.0 必须在 Windows 9x 操作系统下工作。

4. 数据库系统

数据库系统(database system)是指引进数据库技术后的整个计算机系统。它包含硬件系统、系统软件、数据库应用系统和相关人员。其中系统软件包括操作系统、数据库管理系统;数据库应用系统是为特定应用开发的数据库应用软件;相关人员包括数据库管理员(database administrator,简称 DBA)、系统分析员、数据库设计人员、应用程序员和最终用户。数据库系统中各部分的关系如图 1-1 所示。

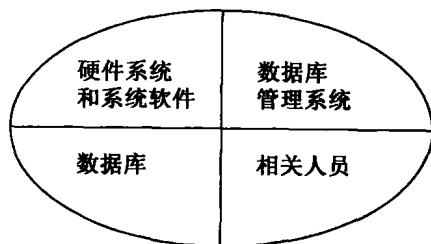


图 1-1 数据库系统组成结构示意图

数据库系统有组织地、动态地存储着有密切联系的数据集合,并通过计算机的软件及硬件资源所组成的系统对其进行统一的管理,因而数据库系统包含了数据库和数据库管理系统。

目前,国内广泛使用的数据库应用系统的实例很多,如大型数据库管理系统中的飞机订票系统、银行系统等等。

1.1.2 数据模型

计算机不能直接处理现实世界中的具体事物,人们必须把具体事物转换成计算机可以处理的数据。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。即模型是现实世界特征的模拟和抽象。在数据库中用数据模型来抽象、表示和处理现实世界中的数据和信息。

数据模型应满足三方面的要求:一是能比较真实地模拟现实世界;二是容易为人所理解;三是便于在计算机上实现。

数据结构、数据操作和完整性约束是构成数据模型的三要素。数据模型主要包括层次模型、网状模型、关系模型等。

1. 层次模型

层次模型的基本结构是一棵倒立的树,与 DOS 中的目录树相似,树的节点表示记录集合,树枝表示记录集合之间的联系。

图 1-2 表示一个学校组织机构的树形结构。其中学校表示根节点,树中仅有一个根,根节点向上没有联系;根的子节点为第二层,根为其子节点的父节点,同一父节点的子节点称为兄弟节点,如系与系之间、专业与专业之间、班与班之间。向上只有一个联系,而向下可以有多个联系的节点,称为中间节点,如二系、 N 系、二系中的专业 N 。向下没有联系的节点称为叶节点,如一系、三系、专业 1、专业 2、一班、二班等。

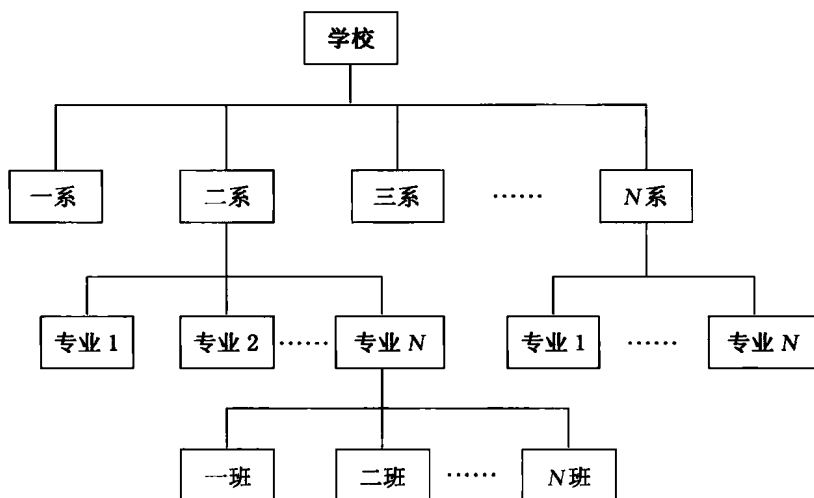


图 1-2 层次模型

满足以下两个条件的数据模型称为层次模型：

- (1) 有且仅有一个节点无父节点,这个节点称为根节点；
- (2) 其他节点有且仅有一个父节点。

层次模型对具有一对多层次关系的描述非常自然、直观、容易理解,这也是采用层次模型的层次型数据库的突出优点。但对于现实中的许多问题,层次型数据库并不能很好地描述其中的实体和实体(即模型中节点和节点)之间的关系。例如,在图 1-2 中所示的数据模型中,如果询问二系中的专业 2 和 N 系中的专业 1 之间的联系,则该模型就不能直接回答,因为层次模型构成单向树,在不同系的各个专业之间并没有直接的路径相通。这也是层次

模型的最大缺点。

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。网状模型是一个网络,是层次模型的拓展。如图 1-3 所示,图中描述了一个学校的教学实体,其中老师、学生两个节点无父节点,课程、成绩、教务处有两个以上的父节点,它们交织在一起形成网状关系,也就是说,一个节点可能对应多个节点。

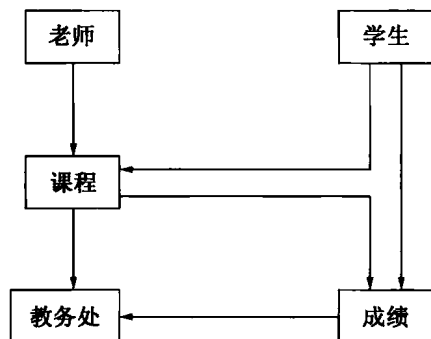


图 1-3 网状模型

满足以下两个条件的数据模型称为网状模型：

- (1) 允许一个以上的节点无父节点;
- (2) 一个节点可以有多于一个的父节点。

层次模型与网状模型的主要区别在于:层次模型中从子节点到父节点的联系是唯一的;网状模型中从子节点到父节点的联系则不是唯一的。网状模型两节点间的联系可以是多对多的联系,且兄弟节点到父节点的联系不是唯一的。

采用层次模型和网状模型的数据库在数据库技术中被称为第一代数据库。

3. 关系模型

关系模型是以数学理论为基础而构造的数据模型,它把数据组织成满足一定条件的二维表形式,这个二维表就是关系。用二维表结构来表示实体及实体之间联系的模型称为关系模型,如表 1-2 所示。

表 1-2 学生情况表

学号	姓 名	性别	籍贯	出生日期	团员	入学成绩	系别
0123	马文广	男	陕西	10/07/1988	T	460.0	计算机系
1243	王 伟	男	甘肃	05/23/1987	T	435.6	电子系
0032	张丽美	女	四川	09/08/1985	F	388.0	计算机系
0065	孙小伟	男	陕西	01/04/1988	T	444.5	外语系
2314	黄 妮	女	江苏	02/23/1987	T	498.8	电子系
0143	王为国	男	甘肃	04/07/1986	F	343.0	计算机系
1023	李佳丽	女	四川	12/30/1985	T	445.6	外语系

在数据库中,满足下列条件的二维表的数据模型为关系模型:

- (1) 每一列中的分量是类型相同的数据;
- (2) 行与列的顺序可以是任意的;
- (3) 表中的分量是不可再分割的最小数据项,即表中不允许有子表;
- (4) 每一列的名称不能相同;
- (5) 表中的任意两行不能完全相同。

关系数据模型中,无论实体还是实体之间的联系都用关系来表示,关系模型中的数据逻辑结构是一张二维表,它由行和列组成;一个关系对应一张二维表,表中一列表示实体的一项属性,称为一个字段;表中的一行包括了一个实体的全部属性值,称为一个记录。

关系数据库因其严格的数学理论、使用简单灵活、数据独立性强而发展十分迅速,目前已成为具有主导地位的数据库形式。采用关系模型的数据库被称为第二代数据库。

1.1.3 面向对象数据库

在数据库技术的研究和发展中,被广泛应用的是面向对象数据库管理系统(object oriented database system,简称 OODS)。面向对象是一种新型的程序设计方法,它能够处理复杂的数据结构,而且,使用面向对象程序设计方法编写的程序更近于自然,便于理解,有利于软件的扩充和再利用。下面简单介绍其基本概念。

1. 对象

所谓对象(object)可以是任何的具体事物。例如,Windows 9x 的桌面、命令按钮、标签等,它们都受 Windows 9x 操作系统的内部程序控制,都可作为对象。又如表 1-1 和表 1-2 所示的实体就是一个表对象,而在表对象中的学号、姓名等也是一个表对象中的另一些对象。对象包括数据和过程,每个对象都有其对应的属性和方法。

2. 对象的属性

能够定义的对象性质、特征和行为,称为该对象的属性(property),如对象的名称、大小、颜色等。对象属性大部分可能不需要设置,只需使用它们隐含的设置就行了,有些需要用户根据需要来设置(通过专门的属性窗口进行设置,或在程序中设置)。有时同一种对象的属性,在不同的地方,需要设置为不同的值。例如对象上的文字,有时需要设置为黑色,有时需要设置成红色。

3. 事件

事件(event)是由对象识别和响应的某些操作,是一些特定的预定义活动,可由用户或系统启动。例如程序由硬盘读入内存就是系统引起的事件;单击或移动鼠标则是由用户操作引起的事件。在多数情况下,事件都是由用户操作而引起的。

发生一个事件时,控制对象被激活并作出相应的响应,称为事件的过程。到底作出什么响应,由事先为该对象编写的一段程序确定。若这一事件不发生,则这段程序就不会运行。如果没有为该事件编写程序,即使事件发生也不会有任何反应。在不同的面向对象程序设计软件中,事件的格式和程序代码是不相同的。

4. 对象的方法

对象的方法(method)是指对象所具有的功能。例如用鼠标单击 Windows 9x 的“开始”命令按钮后,会弹出开始菜单,并可继续使用。

“事件”和“方法”有相似之处,都是为了完成某个任务。但也有不同之处:

(1) 对象的事件集合基本上是固定的,用户不能创建,而方法可以由用户创建。

(2) 事件的程序代码由用户编写,该事件发生时完成什么任务,完全取决于所编写的程序代码。而方法的程序代码则是系统内置的、固定的,一般不由用户来编写(用户自己创建的方法除外),任何时候调用都是完成同一个任务。

5. 类

类(class)是对一种对象的归纳和抽象,对象的属性、事件和方法程序都在定义类时被指定。类和对象关系密切,但并不相同。类包含了有关对象的特征和行为信息,是对象的模板。例如二维表是一个类,而表 1-2 则是这个类的一个实例,也是一个对象。

6. 子类

子类(subclass)是以其他类定义为起点而给某一对象所建立的新类。一个子类可以拥有派生它的类的全部功能,即具有继承性,并且在此基础上,可添加其他属性或功能。对每个类而言,派生该类的类为父类,被派生的类为子类。此外,由于继承性的存在,如果某个类中发现问题,就不需要逐个修改它的子类,只需对这个类本身做适当的修改即可,子类将继承任何对父类所做的修改。例如表格是一个类,二维表是它的一个子类(除了二维表,还有非二维的复杂表格),二维表是从表格这个类中派生出来的。

1.2 关系数据库

1.2.1 关系数据库的基本概念

由关系模型建立的数据库称为关系数据库。为了进一步了解关系数据库,先介绍关系模型中的一些基本概念。

1. 关系

一个关系就是一张二维表格,每个关系有一个关系名。如前所述的表 1-1 学生成绩表和表 1-2 学生情况表。在计算机里,一个关系可以存储为一个文件。在 Visual FoxPro 6.0 中,一个关系就是一个表文件。

关系有三种类型:基本表、查询表和视图表。基本表是实际存在的表,它是实际存储数据的逻辑表示。查询表是查询结果对应的表。视图表是由基本表或其他视图表导出的表,是虚表,不对应实际存储的数据。

2. 元组

二维表格中水平方向的行称为元组,元组又称为记录。如表 1-1 和表 1-2 中每一行称为一个元组,也叫记录,它们分别有 7 个元组(或记录)。

3. 属性

二维表格中垂直方向的列称为属性,属性又称为字段;每一列有一个属性名,也称为字段名。定义一个属性(字段)时需要确定它的属性名、该属性的数据类型和宽度。属性值是各个记录的该字段的值。如表 1-1 和表 1-2 中每一列称为一个属性,它们分别有 7 个和 8 个属性。“学号”、“姓名”等就是属性名,而“0123”就是 1 号记录的“学号”属性的值,“马文广”就是 1 号记录的“姓名”属性的值。

4. 域

属性的取值范围称为域,即不同元组对同一个属性的取值范围。如表 1-1 中数学、英语、VFP 程序设计、电子技术和体育的域为 0~100,表 1-2 中的性别只能取“男”和“女”两者之一。

5. 关系模式

对关系(二维表)中全部数据的逻辑结构的描述,称为关系模式。例如表 1-2 的关系模式为:学生情况表(学号,姓名,性别,籍贯,出生日期,团员,入学成绩,系别),其中,学生情况表是关系名,学号、姓名等为属性名。

一个关系模式对应一个关系的结构,可以将关系定义为元组的集合,在不至于引起混淆的情况下,往往将关系模式和关系系统称为关系。

6. 主关键字和外部关键字

主关键字:在关系数据库的表中,某个字段(或几个字段的组合)每取一个值,可以确定唯一的一行(记录),该字段(或几个字段的组合)就称为主关键字,又称主码。例如表 1-2 学生情况表中的学号就能唯一地确定每个记录,可作为主关键字;而性别、籍贯、出生日期、团员、入学成绩、系别等都可能具有相同值,因此不能用它们作为主关键字。确定主关键字的条件是不能有重复值,不能有空(NULL)值。

外部关键字:在多个表中,如果 B 表的某个字段(或几个字段的组合)是 A 表中的主关键字,则在 B 表中该字段(或几个字段的组合)称为外部关键字。例如表 1-2 学生情况表中以学号为主关键字,与表 1-1 学生成绩表建立起关联,则表 1-1 中的学号字段称为外部关键字。

一般来说,外部关键字与相应的主关键字具有相同的名称和类型。

1.2.2 关系运算

对关系数据库进行查询时,若要找到用户需要的数据,就需要对关系进行一定的关系运算。关系的基本运算有两种:一种是传统的集合运算,即并、差、交等;另一种是专门的关系运算,即选择、投影、联接等。有些查询需要几个基本运算的组合,经过若干个步骤才能完成。关系运算的操作对象是关系,运算的结果仍为关系。

1. 传统的集合运算

(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。

例如,有两个结构相同的学生关系,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的操作就是这两个关系的并。

(2) 差

设有两个相同结构的关系,两个关系差的结果是由属于第一个关系但不属于第二个关系的元组组成的集合,即差运算的结果是从第一个关系中去掉也属于第二个关系的所有的元组。

例如,设有参加计算机小组的学生关系,参加航模小组的学生关系,求参加了计算机小组,但没有参加航模小组的学生,就应当进行差运算。

(3) 交

两个具有相同结构的关系,它们的交是由既属于第一个关系,又属于第二个关系(即两个关系的公共部分)的元组组成的集合。

例如,设有参加计算机小组的学生关系,参加航模小组的学生关系,求既参加计算机又参加航模小组的学生,就应当进行交运算。

2. 专门的关系运算

(1) 选择

在关系中找到满足某些条件的元组的操作称为选择,也就是在二维表中选择满足指定的条件的行。通常选择的条件以逻辑表达式给出,使逻辑表达式的值为真的元组将被选择。例如在表 1-2 学生情况表中找出系别为“计算机系”的学生,或者找出“入学成绩”高于“450”分的学生等。

(2) 投影

在关系模式中选择若干个属性组成新的关系称为投影,也就是在二维表中对列的运算。经过投影运算可以得到一个新关系,新关系往往比原关系的属性个数要少,或者属性的排列顺序有所不同。例如在表 1-2 学生情况表中仅需要学生的学号、姓名、性别、籍贯、入学成绩这几个字段,就可以使用投影运算来实现。

(3) 联接

联接是关系的横向结合。联接过程是通过联接条件来控制的,联接条件中的属性称为联接属性,两个关系中的联接属性应该有相同的数据类型。

联接运算中有两种基本联接:一种是等值联接,一种是自然联接。当联接条件中的关系运算符为“=”时,表示等值联接;自然联接是一种特殊的联接,它要求两个关系中进行比较的分量必须是相同的属性组,并且在结果中把重复的属性列去掉。

例如由表 1-1 和表 1-2 重新生成一个新的二维表,该表中包含表 1-1 的学号、姓名、数学、英语、VFP 程序设计、电子技术、体育和表 1-2 的性别、籍贯、出生日期、系别的操作就应该采用联接操作。

选择和投影运算的操作对象是一个关系,相当于对一个二维表进行操作,联接运算需要把二个关系作为操作对象,相当于对两个二维表进行拼接。在关系数据库管理系统中,对于上述的关系运算都有相应的操作命令,实际的操作通常是几种操作的综合应用。

1.2.3 关系的完整性

关系模型的完整性(integrity)规则是对关系的某种约束条件。关系模型中有三种完整性约束:实体完整性、参照完整性和用户定义的完整性。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,被称为关系的两个不变性,应该由关系系统自动支持。

1. 实体完整性

是指主关键字的值在关系中必须是非空且必须是唯一的。实体完整性规则规定基本关系中组成主关键字的所有属性都不能取空值(如果主关键字是由多个属性组成)。例如,表 1-1 学生成绩统计表(学号,姓名,成绩)中,“学号”为主关键字,因此“学号”不能取空值,而不是整体不为空。如果以“学号”和“姓名”两个字段的组合作为主关键字,则“学号”和“姓名”都不能取空值,而不仅是“学号”不能取空值。

2. 参照完整性

参照完整性是定义外部关键字与主关键字之间引用的规则。引用的时候必须取基本表中已经存在的值。

参照完整性是指一个表中外部关键字的值必须是相应数据库中的其他关系的主关键字值之一,或为空。例如,学生成绩统计表(学号,姓名,成绩)的学号字段的每一个分量必须是学生关系的学号字段的分量之一。

3. 用户定义的完整性

不同的关系数据库系统根据其应用环境的不同,往往还需要一些特殊的约束条件,用户定义的完整性就是针对某一具体关系数据库的约束条件,用户定义的完整性又称为域完整性。例如,单科成绩的取值范围通常是 0~100 之间,性别的取值范围也只能是“男”或“女”两个值中的一个。

1.3 Visual FoxPro 系统特点与工作方式

1.3.1 Windows 版本数据库的特点

Visual FoxPro 是在 FoxPro for Windows 的基础上发展起来的,其核心是可视化程序

设计。Visual FoxPro 使用户对数据的组织、定义数据库及相关规则、建立应用系统变得非常方便;可视化的工具或向导,使用户能很快地建立表单、查询和报表等。大量使用向导、设计器、生成器等界面操作工具,把传统的命令执行方式扩充为以界面操作为主、命令方式为辅的交互执行方式,把单一的面向过程的结构化程序设计扩充为既有结构化设计,又有面向对象程序设计的可视化程序设计。如果能充分利用 Visual FoxPro 6.0 提供的集成环境、强大的面向对象的编程工具、客户机/服务器功能和对 OLE 与 ActiveX 的支持,可以使用户在建立复杂应用系统时更为简单而方便,使 Visual FoxPro 成为当今最流行的功能强大的数据库系统。

1. 强大的查询与管理功能

(1) Visual FoxPro 6.0 拥有约 500 条命令、200 余种函数,其功能空前地强大。加上 Rushmore 快速查询技术,可将查询响应时间从数小时或数分钟降低到数秒,可以显著地提高查询速度。

(2) Visual FoxPro 6.0 可以允许多个开发者同时工作,如果是几个开发者开发同一个应用程序,可以使用 Visual FoxPro 6.0 同时访问数据库组件的功能。数据库容器允许几个用户在同一个数据库中同时创建或修改对象。利用“数据库设计器”可以迅速更改数据库中对象的外观。

(3) 项目管理器对数据、文档、源代码、类库等资源进行集中高效的管理,若要跟踪或保护对源代码的更改,还可以使用带有“项目管理器”的源代码管理程序,进一步加强了 Visual FoxPro 的功能。此外,还可以借助“项目管理器”创建和集中管理应用程序中的任何元素,并使用所有向导、生成器、工具栏和其他易于使用的工具。“项目管理器”提供了一个进行集中管理的环境,可以对系统本身定义工具栏,也能够为编写的应用程序定义工具栏,使用户可以快速开发应用程序。

(4) Visual FoxPro 6.0 添加了新的“应用程序向导”,提供了新的 ProjectHook 对象,并改进了应用程序框架功能,使应用程序更有效率。同时又添加了一些功能来增强开发环境,以便更容易地向应用程序中添加有效的功能。

2. 引入了数据库表的新概念

在 Visual FoxPro 以前的数据库系统中,一个数据库文件就是一张二维表,各个库文件之间的联系是在使用时临时建立的。而 Visual FoxPro 引入了数据库表的新概念,一张二维表只是一个数据表,把相互之间有联系的数据表集中在一起,成为一个数据库。对所有的数据库表,在建表时就同时定义它与库内其他表之间的关系,使 Visual FoxPro 建立的数据库表更加符合数据库的实际要求,也方便了用户随后对这些表的使用。

3. 扩大了对 SQL 语言的支持

SQL 语言是关系数据库的标准语言,查询功能强大,使用灵活。在 Visual FoxPro 中,SQL 命令已扩充为 8 种,这就加强了 Visual FoxPro 语言的功能,同时也为 Visual FoxPro 的使用者提供了一个学习、熟悉 SQL 语言的机会。

4. 大量使用可视化的界面操作工具

Visual FoxPro 提供了向导、设计器、生成器等三类界面操作工具,达 40 多种。使用者可以利用这些可视化的界面操作工具,根据不同界面的不同要求,在相应的界面中选择其中的选项或者回答其中的问题,即可进入下一个步骤,如此进行下去,就可以完成相应的任务。

另外, Visual FoxPro 的设计器普遍配有工具栏和弹出式的快捷菜单, 大多数设计器也可提供快捷菜单, 其中包含最常用的菜单选项, 以供用户随时调用。通过对界面的改造, 如今的 Visual FoxPro 6.0 在向导、生成器、工具栏和设计器的帮助下, 使应用系统的开发变得十分简单。

5. 支持面向对象的程序设计

允许用户对“对象”和“类”进行定义, 并编写相应的代码。用户可以在类的基础上定义自己的类和子类, 由于子类对类有继承性, 这样就减少了编程的工作量, 加快了软件开发的过程。

6. 通过 OLE 实现应用集成

Visual FoxPro 可与包括 Word 与 Excel 在内的其他应用软件数据共享。在 Visual FoxPro 6.0 中, 只要打开相应的文件, 就会出现转换对话框, 可以把其他数据移到 Visual FoxPro 6.0 表中, 如果有电子表格或文本文件中的数据, 比如 Microsoft Excel 或 Word 文件, Visual FoxPro 6.0 可以方便地实现数据共享。

7. 支持网络应用

支持客户机/服务器结构, 既可访问本地计算机, 也可支持对服务器的浏览。对于来自本地或远程的多个数据库表的不同种数据, Visual FoxPro 支持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。在多用户环境中, 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等。

1.3.2 Visual FoxPro 的启动和退出

1. 启动 Visual FoxPro

(1) 单击“开始”按钮, 系统弹出“开始”菜单, 从“开始”菜单中选择“程序”;

(2) 从“程序”子菜单选择“Microsoft Visual FoxPro 6.0”, 系统立即启动 Visual FoxPro, 出现如图 1-4 所示的 Visual FoxPro 6.0 窗口;

(3) 用户也可以将 Visual FoxPro 6.0 选项设置成快捷方式图标, 放在桌面系统上, 启动时, 只要双击该图标即可。设置快捷方式图标的方法很简单, 在 Windows 98 下, 单击“开始”菜单, 选择“程序”并找到“Microsoft Visual FoxPro 6.0”选项, 直接用鼠标将该项拖动到桌面上即可。

2. 退出 Visual FoxPro

退出 Visual FoxPro 6.0 可以用下面五种方法:

- 选择“文件”菜单中的“退出”命令。
- 单击工作窗口标题栏右上角的“关闭”按钮。
- 在命令窗口中键入“Quit”命令。
- 双击工作窗口左上角的控制菜单图标。
- 直接按[Alt]+[F4]键。

1.3.3 Visual FoxPro 的工作界面

Visual FoxPro 6.0 是一种可视化的界面操作工具, 因此它的工作界面有很多, 这里不能一一列举, 在此我们只简单介绍其主工作界面。启动 Visual FoxPro 6.0 后出现的工作界