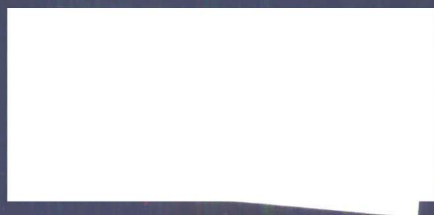


高职高专计算机系列教材
Gaozhi Gaozhuan Jisuanji Xilie Jiaocai

微机数据库程序设计及应用 ——FoxPro for Windows

沈祥玖 主编 岳国英 副主编
沈祥玖 岳国英 刘於勋 解永刚 尹 涛 编著



高等教育出版社

HIGHER EDUCATION PRESS



微机数据库程序设计及应用

—— FoxPro for Windows

沈祥玖 主编 岳国英 副主编
沈祥玖 岳国英 刘於勋 解永刚 尹涛 编著

高等教育出版社

内 容 提 要

本书根据国家教育委员会高等工程专科学校非计算机专业计算机课程教学基本要求而编写。主要针对目前流行的 FoxPro for Windows 2.5, 2.6, 3.0 几个版本。内容包括: 数据库概述, FoxPro 简介, FoxPro 基础知识, 系统菜单的使用, 表的基本操作, 程序设计, 内存变量及文件操作, 多重表操作, 输入输出与界面技术, 开发工具, 对象的链接与嵌入, 应用举例等。本书可作为高等专科学校非计算机专业计算机程序设计课程的教材, 也可作为大中专及各类微机培训班、计算机知识与应用能力等级考试的教材或教学参考书。

图书在版编目(CIP)数据

微机数据库程序设计及应用: FoxPro for Windows/沈
样玖主编. —北京: 高等教育出版社, 1997. 12(2001 重印)
ISBN 7-04-006209-7

I. 微… II. 沈… III. 数据库管理系统, FoxPro-程序
设计 IV. TP312

中国版本图书馆 CIP 数据核字(97)第 21850 号

出版发行 高等教育出版社

社 址 北京市东城区沙滩后街 55 号

邮政编码 100009

电 话 010—64054588

传 真 010—64014048

网 址 <http://www.hep.edu.cn>

经 销 新华书店北京发行所

印 刷 北京市朝阳区北苑印刷厂

开 本 787×1092 1/16

版 次 1997 年 11 月第 1 版

印 张 20.5

印 次 2001 年 4 月第 5 次印刷

字 数 500 000

定 价 19.80 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

当今信息时代,以计算机技术为代表的高新技术飞速发展,计算机的应用向各个领域迅速渗透,了解和使用计算机已成为人们参与社会活动所应具备的基本能力。

信息管理是计算机应用的主流,数据库技术是这个领域中普遍采用的技术。本书介绍 20 世纪 90 年代最新关系型数据库管理系统 FoxPro。FoxPro 由 Fox Software 公司于 1989 年首次推出,并在不到 10 年的时间内,经过了几次版本升级。本书主要针对目前流行的 FoxPro for Windows 2.5, 2.6, 3.0 几个版本而编写。

FoxPro 与 DBASE、FoxBASE 兼容,但在功能、速度、用户界面等方面均有大的突破和改进。如简易快捷的菜单及按钮操作,采用了对数据的索引式存取加以优化的技术和新的快速查询方法,提供了菜单生成器、项目管理等开发工具,能够将应用程序编译为 .exe 文件,具有高级外部程序接口等。FoxPro 所具有的优异性能,使它受到广大用户的青睐,应用前景十分广阔。

本书可作为高等专科学校非计算机专业计算机程序设计课程的教材,也可作为大中专及各类微机培训班的教材或教学参考书,教学参考学时为 80 学时,其内容分为数据库概述、FoxPro 简介、FoxPro 基础知识、系统菜单的使用、表的基本操作、程序设计、内存变量及文件操作、多重表操作、输入输出与界面技术、开发工具、对象的链接与嵌入、应用举例等,共 12 章。第 1~9 章为教学基本内容,介绍数据库的基础知识、FoxPro 的运行环境、操作方法、程序的基本结构等,体现打好基础,面向应用的指导思想,学时数控制在 60 学时左右。第 10、11 章为提高部分,主要介绍面向对象的程序设计,突出以应用开发为主要内涵的程序设计手段,学时数控制在 10 学时左右。最后一章为应用程序实例,系统介绍应用软件的开发过程和方法。

本书由国家教育委员会高等工程专科学校计算机课程教学指导委员会组织全国部分高等专科学校有经验的教师编写,由沈祥玖任主编,岳国英任副主编;全书由赵景林主审。编写分工如下:第 1、6、12 章由岳国英编写;第 2、5 章由刘於勋编写;第 3、4 章和第 9 章的第 4 节及附录由解永刚编写;第 7、8 章和第 9 章的 1、2、3 节由尹涛编写;第 10、11 章由沈祥玖编写,最后由沈祥玖统稿。

限于水平和经验,教材中难免出现差错和疏漏,敬请读者指正。

作者

1997 年 9 月

目 录

第一章 数据库概述.....	1
1.1 数据库的基本概念.....	1
1.1.1 数据和信息(Data & Information).....	1
1.1.2 数据库系统(DBS).....	1
1.2 数据模型.....	3
1.2.1 层次模型.....	3
1.2.2 网状模型.....	4
1.2.3 关系模型.....	5
1.3 关系数据库.....	6
1.3.1 基本概念.....	6
1.3.2 关系操作.....	7
1.3.3 SQL 语言简介.....	9
1.3.4 关系数据库的规范化.....	10
习题一.....	10
第二章 FoxPro 简介.....	11
2.1 FoxPro 的发展过程.....	11
2.2 Windows 95 的基本知识和使用.....	12
2.2.1 Windows 95 操作系统的特色.....	12
2.2.2 中文 Windows 95 的启动.....	13
2.2.3 程序管理器及鼠标指针的功能.....	13
2.2.4 中文 Windows 95 应用软件的启动与切换.....	15
2.2.5 Windows 95 系统的退出.....	16
2.3 FoxPro 的主要特点及性能指标.....	17
2.3.1 FoxPro 系统的主要特点.....	17
2.3.2 FoxPro 的性能指标.....	18
2.4 文件类型.....	20
2.5 FoxPro 系统的运行环境.....	21
2.6 FoxPro 系统的安装、启动及退出.....	21
2.6.1 FoxPro 系统的安装.....	21
2.6.2 FoxPro 系统的启动.....	24
2.6.3 FoxPro 系统的退出.....	24
2.7 FoxPro 系统的运行方式.....	25
习题二.....	26
第三章 FoxPro 基础知识.....	27
3.1 数据类型.....	27
3.1.1 数据类型.....	27

3.1.2 常量	28
3.1.3 变量	28
3.2 表达式	29
3.2.1 数值表达式	29
3.2.2 字符表达式	29
3.2.3 日期型表达式	30
3.2.4 关系表达式	30
3.2.5 逻辑表达式	31
3.3 函数	31
3.3.1 数值函数	32
3.3.2 字符函数	33
3.3.3 转换函数	36
3.3.4 测试函数	38
3.3.5 日期函数	39
3.4 语法约定	40
习题三	42
第四章 系统菜单的使用	43
4.1 菜单系统	43
4.2 File(文件)菜单	44
4.3 Edit(编辑)菜单	48
4.4 Database(表/数据库)菜单	52
4.5 Record(记录)菜单	55
4.6 Program(程序)菜单	57
4.7 Run(运行)菜单	58
4.8 Text(文本)菜单	60
4.9 Window(窗口)菜单	60
4.10 Help(帮助)菜单	61
习题四	61
第五章 表的基本操作	62
5.1 表文件	62
5.1.1 定义表结构	62
5.1.2 表文件的建立	64
5.1.3 显示表结构	70
5.1.4 修改表结构	71
5.1.5 表文件的打开与关闭	73
5.2 增加记录	75
5.3 记录的定位	79
5.4 记录的显示	83
5.5 修改记录	85
5.5.1 BROWSE 命令	85
5.5.2 REPLACE 命令	93
5.6 删除记录	95

5.6.1 DELETE 命令	95
5.6.2 RECALL 命令	96
5.6.3 PACK 命令	96
5.6.4 ZAP 命令	97
5.7 排序、索引和查询	97
5.7.1 排序	97
5.7.2 索引和查询	100
5.8 数据统计	104
5.8.1 数值统计	104
5.8.2 财务统计	107
5.8.3 分类统计	108
习题五	110
第六章 程序设计	112
6.1 命令文件	112
6.1.1 命令文件的格式	112
6.1.2 命令文件的建立与修改	112
6.1.3 命令文件的编译	113
6.1.4 命令文件的执行	114
6.2 程序中常用的命令	114
6.2.1 输入命令	114
6.2.2 输出命令?和??	117
6.2.3 常用的系统状态设置命令	118
6.2.4 其他命令	118
6.3 结构化程序设计	119
6.3.1 结构化程序设计概述	119
6.3.2 结构化程序的基本结构	120
6.3.3 结构化程序设计的方法	120
6.4 顺序结构程序设计	121
6.5 选择结构程序设计	123
6.5.1 IF ... ELSE ... ENDIF 语句	123
6.5.2 DO CASE ... ENDCASE 语句	126
6.6 循环结构程序设计	128
6.6.1 DO WHILE ... ENDDO 循环	129
6.6.2 SCAN ... ENDSCAN 语句	132
6.6.3 FOR ... ENDFOR 语句	134
6.7 子程序、过程及过程文件	139
6.7.1 子程序和过程	139
6.7.2 过程文件	147
6.7.3 自定义函数	150
习题六	152
第七章 内存变量及文件操作	154
7.1 内存变量操作	154

7.1.1 内存变量的建立	154
7.1.2 内存变量的显示	155
7.1.3 内存变量的释放	156
7.1.4 内存变量的保存	157
7.2 数组的应用	157
7.2.1 数组的定义	157
7.2.2 数组的赋值	158
7.2.3 数组与表间的数据传递	159
7.2.4 数组函数	165
7.2.5 数组应用举例	167
7.3 内存变量文件	167
7.4 表的复制	169
7.4.1 用当前选定表的内容创建新表	169
7.4.2 从其他表转入记录	170
7.5 文件转换	170
7.5.1 表文件转换成文本文件	170
7.5.2 文本文件转换成表文件	172
7.6 结构描述文件	173
7.6.1 生成结构描述文件	173
7.6.2 由结构描述文件创建一个表	174
习题七	174
第八章 多重表操作	176
8.1 工作区的概念	176
8.2 工作区互访	178
8.3 在多表之间建立关系	179
8.3.1 关系的概念	180
8.3.2 按索引关键字表达式建立关系	180
8.3.3 按记录号表达式建立关系	183
8.3.4 表之间关系的分类	184
8.4 多表查询技术	184
8.5 多重表的应用举例	187
8.5.1 表的更新	187
8.5.2 优化数据存储设计	188
8.6 库的基本操作	191
8.6.1 库的概念	191
8.6.2 创建一个数据库	191
8.6.3 打开一个数据库	192
8.6.4 关闭数据库	193
8.6.5 表与数据库	193
8.7 视图操作	195
8.7.1 创建视图	196
8.7.2 创建参数化视图	197
8.7.3 使用视图	198

8.7.4 视图的可更新属性	199
8.8 别名	200
8.8.1 在多个工作区打开同一个表	200
8.8.2 通用性程序设计使用别名	201
习题八	202
第九章 输入/输出与界面技术	203
9.1 窗口设计	203
9.1.1 定义窗口	204
9.1.2 激活窗口	206
9.1.3 窗口的显示	208
9.1.4 隐藏窗口	209
9.1.5 释放窗口	210
9.1.6 从内存中释放窗口	210
9.2 屏幕设计	211
9.2.1 清除屏幕	211
9.2.2 屏幕输出格式设计	211
9.2.3 屏幕数据输入格式设计	212
9.2.4 选项按钮	214
9.2.5 复选框	215
9.2.6 列表框	216
9.2.7 组合框	218
9.3 菜单设计	219
9.3.1 创建菜单系统过程及方法	219
9.3.2 光条菜单设计	220
9.3.3 系统风格的菜单	222
9.3.4 命令按钮	227
9.4 打印报表设计	230
9.4.1 报表格式设计	231
9.4.2 系统报表设计	231
9.4.3 有关打印输出设计命令	234
9.4.4 打印报表程序设计	236
习题九	238
第十章 开发工具	239
10.1 项目管理器	239
10.1.1 项目管理器的功能	240
10.1.2 “项目管理器”窗口的功能按钮	242
10.2 表单生成器	243
10.2.1 使用表单向导创建即用表单	243
10.2.2 使用“属性”窗口对表单添加和修改控制	245
10.2.3 使用“表单控制”工具栏在表单上创建控制	246
10.2.4 使用“调色板”工具栏	249
10.2.5 运行表单	251

10.3 菜单生成器	251
10.4 查询生成器	253
10.4.1 创建一个新的查询文件	253
10.4.2 运行查询和修改查询	256
10.4.3 “查询”菜单	257
10.5 报表生成器	260
10.5.1 使用报表向导创建新的报表	260
10.5.2 使用报表设计器和报表控制工具栏	262
10.5.3 定义报表的页面	264
10.5.4 打印报表	265
10.6 编译工具	266
习题十	267
第十一章 对象的链接与嵌入(OLE)	268
11.1 OLE 有关概念	268
11.2 将 OLE 添加到表中	268
11.3 向表单链接或嵌入数据	271
11.4 OLE 控制概述	272
习题十一	274
第十二章 数据库应用系统开发实例——学生成绩管理系统	275
12.1 系统分析	275
12.1.1 问题的提出	275
12.1.2 需求分析	275
12.2 系统设计	276
12.2.1 功能设计	276
12.2.2 数据库设计	279
12.2.3 编码设计	282
12.2.4 单元测试	289
12.2.5 系统测试	290
习题十二	290
附录一 FoxPro 常用命令	291
附录二 FoxPro 常用函数	310

第一章 数据库概述

近 10 多年来, 由于计算机技术的发展和数据库管理系统的出现, 使得计算机系统的应用从军事和科学计算领域, 逐步扩展到社会的各个领域, 为计算机的应用普及打开了一个崭新的局面。目前各种类型的计算机已经广泛应用于情报检索系统、办公信息系统、银行信息系统、交通信息系统、图书管理系统、人事档案管理系统、财务管理系统等各种企事业管理信息系统中。管理信息系统是计算机应用最广泛的领域, 占计算机应用的 70%~80%。

管理信息系统中, 关键的问题是如何对系统的大量数据进行有效、合理的存储和处理, 为用户准确、快速地提供所需要的信息。数据库技术就是为了解决这一问题而产生和发展起来的。

本章将介绍数据库的基本概念和基本知识, 为读者学习一种具体的数据库管理系统打下基础。

1.1 数据库的基本概念

1.1.1 数据和信息(Data & Information)

数据(Data)是对客观事物属性的描述与记载, 是一些物理符号。信息(Information)是客观世界中各种事物(包括数据)变化、相互作用和特征的反应, 是一个抽象的概念。

数据是信息的具体表现形式, 信息是数据有意义的表现。信息来源于数据, 当数据传递给有关的人, 通过相互作用, 可能会给人带来某种信息。可见, 数据与信息既有联系又有区别, 但有时并不进行区分。例如, 数据处理和信息处理往往是指同一个概念。

数据处理(Data Processing)是指对数据进行整理、存储、分类、排序、统计、加工和分析, 以得到所需信息的过程。

在计算机科学中, 所有能够输入到计算机中处理的对象, 都是计算机的“数据”。例如: 数值、文字、声音、图像等。人类进入信息化社会后, 信息处理的内容与范围将越来越广泛, 越来越复杂。

1.1.2 数据库系统(DBS)

一、计算机数据管理技术的发展

自 20 世纪 40 年代后期电子计算机诞生以来, 利用计算机进行数据处理经历了如下三个阶段。

1. 自由管理阶段

自由管理阶段是用计算机进行数据处理的初级阶段。此阶段计算机主要用于数值计算。数据处理的性质只是使用计算机代替手工劳动。

这一阶段中, 计算机硬件功能较弱, 输入输出设备简单, 外部存储介质也只有容量极低的磁带、卡片和纸带等。软件方面既没有操作系统, 也没有文件管理能力, 软件只能处理简

单的输入输出操作。计算机对数据的管理,从规定数据的输入输出格式到数据的逻辑结构,从存取方法到数据的物理结构,完全由程序员自由安排。

这一阶段的特征是:数据不能独立于程序,存取操作复杂繁琐,而且不能长期保存;一组数据对应一个程序,数据之间、程序之间是相互独立的,因此,程序之间会出现数据冗余,数据不能共享。

2. 文件系统阶段

20 世纪 60 年代中后期,大容量磁盘组和灵活的软磁盘成为主要的外部存储介质,输入输出功能大大增强。软件方面出现了操作系统,其中包含用于管理数据的文件系统。

文件(File)是存储在计算机外部存储介质(主要是磁盘)上的数据的有序集合,并以指定的文件名与扩展名登录在磁盘目录中。文件中的数据以“记录”形式存储;记录由若干相关的数据项组成,若干相同性质的记录集合便构成一个文件。数据以文件形式长期存放在外部存储介质上,应用程序通过文件系统对文件中的数据进行存取、查询、修改、删除等处理,这就使得应用程序和数据都有一定的独立性,数据的冗余也有所减少。用户设计应用程序时,只需要考虑数据的逻辑定义和物理特征,按规定的组织方式建立文件和使用文件,基本上不用考虑数据的物理存储问题。

这一阶段的不足在于数据之间缺乏有机的联系,所以不同程序数据的冗余度仍然很大,不能充分实现数据共享;另外,数据缺乏充分的独立性,主要反映在数据结构改变,应用程序也必须做相应的改变;应用程序对数据的处理也比较繁琐,这就使得数据的管理与控制不方便。

3. 数据库管理阶段

20 世纪 70 年代出现的数据库技术解决了文件系统管理方式存在的不足。数据库中的数据以不同的组合形式,同时为多个应用程序或用户所共享;数据库系统还提供了一整套管理和控制数据的操作命令,并对数据完整性、唯一性、安全性、保密性提供了有效的管理手段。数据库技术是一种比较完善、数据管理能力较强的计算机管理技术。当前,数据库系统已成为信息处理的核心技术,数据库技术的最新成果也就反映了数据处理的当前水平。

二、数据库系统

数据库系统(Data Base System, 简称 DBS)通常是指由数据库、数据库管理系统和应用程序组合而成的具有数据库管理功能的计算机系统,其层次结构如图 1.1 所示。

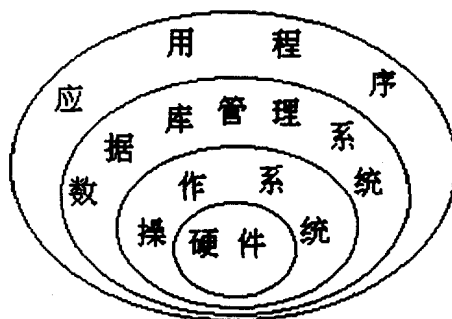


图 1.1 数据库系统层次结构图

1. 硬件(Hardware)。硬件一般指由电子部件和机电装置组成的计算机实体。包括中央处理器、存储器、输入设备和输出设备等。

2. 操作系统(Operating System, 简称 OS)。操作系统是实现了对计算机的全部软件、硬件资源的控制和管理, 使计算机能够自动、协调、高效地工作的一组软件。

操作系统是硬件的第一级扩充, 只有通过操作系统才能使用计算机系统。

3. 数据库管理系统(Data Base Management System, 简称 DBMS)。数据库管理系统是对数据库进行管理的系统软件, 它执行用户发出的有关数据操作的命令, 统一管理数据库中的数据资源。一般地说, 它应该有数据库定义、数据的操作(包括输入、输出、修改、检索、统计等)、数据库的运行管理等功能。DBMS 在 OS 的支持下既面向用户, 又面向整个数据库系统。

4. 数据库(Data Base, 简称为 DB)。简单地说, 数据库是一个提供数据的基地, 其中数据按一定的格式存放在计算机的存储介质上, 以供用户访问。也就是说, 数据库是长期存储在计算机系统中、有组织、可共享的数据的集合。数据库中的数据应按一定的数据模型组织、描述和存储, 具有较小的冗余度, 较高的数据独立性和易扩展性, 并且可以为各种用户共享。

5. 用户应用程序是用户自己为处理和享用数据库中存储的数据而编写的程序。DBMS 通常都包含编程语言, 为用户提供一系列与高级语言类似的语句, 用户可以直接使用这些语句来编写应用程序。

1.2 数据模型

对于模型这一概念, 大家并不陌生, 特别是一些具体模型, 在日常生活中时常可见。例如, 地形图沙盘、按一定比例缩小的建筑模型、精致的航模飞机和军舰等都是人们熟悉的具体模型。看到这些模型人们便会联想起现实生活的事物。这里要讨论的数据模型(Data Model)也是一种模型, 它是数据特征的抽象, 描述的是数据的共性。

数据模型是我们对数据进行抽象表示或者说是模型化的工具, 应用的目的不同, 对数据进行抽象的工具也就不同, 也就是说模型也不相同。根据模型应用的不同目的, 可以将其分为两类: 一类是概念模型, 或称为“信息模型”, 是按用户观点将数据模型化; 另一类是数据模型, 是按计算机观点对数据建模。

由于篇幅所限, 这里只讨论第二类模型。这类模型直接面向数据库中数据的逻辑结构。目前, 数据库领域中最常用的数据模型有以下三种:

- 层次模型 (Hierarchical Model)
- 网状模型 (Network Model)
- 关系模型 (Relational Model)

下面我们来讨论这三种基本数据模型。

1.2.1 层次模型

若用图来表示, 层次模型是一棵倒立的树。在数据库中, 满足以下两个条件的数据模型称为层次模型:

- (1) 有且仅有一个结点无父结点, 这个结点称为“根结点”;

(2) 其他结点有且仅有一个父结点。

在层次模型中，结点层次(Level)从根开始定义，根为第一层，根的子结点称为第二层，根称为其子结点的父结点，同一父结点的子结点称为兄弟结点。

图 1.2 给出了一个学校行政机构的层次模型。

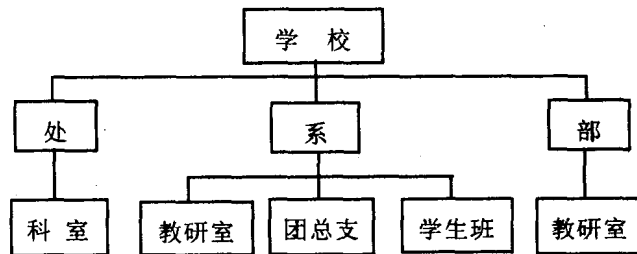


图 1.2 简单的层次模型

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，这是层次数据库的突出优点。

层次数据库采用层次模型作为数据的组织形式。典型的层次数据库管理系统是 1968 年 IBM 公司推出的 IMS 系统。

1.2.2 网状模型

若用图表示，网状模型是一个网络。在数据库中，满足以下两个条件的数据模型称为网状模型：

- (1) 允许一个以上的结点无父结点；
- (2) 一个结点可以有多个父结点。

由于在网状模型中子结点与父结点的联系不是唯一的，所以，要为每个联系命名，并指出与该联系有关的父结点和子结点。图 1.3 给出了一个抽象的简单的网状模型。

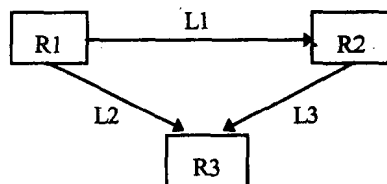


图 1.3 简单的网状模型

R1 与 R2 之间的联系被命名为 L1，R1 与 R3 之间的联系被命名为 L2，R2 与 R3 之间的联系被命名为 L3。其中 R1 为 R2、R3 的父结点，R2 也是 R3 的父结点。

图 1.4 给出了一个具体的简单的网状模型：学生—选课—课程的网状模型。

网状模型允许结点无父结点或有一个以上的父结点，从而构成了比层次结构复杂的网状结构。

网状数据库采用网状模型作为数据的组织方式。网状数据库管理系统的典型代表是 20 世纪 70 年代美国的数据系统研究会 CODASYL(Conference On Data System Language)下属的数据库任务组 DBTG(Data Base Task Group)提出的 DBTG 系统。

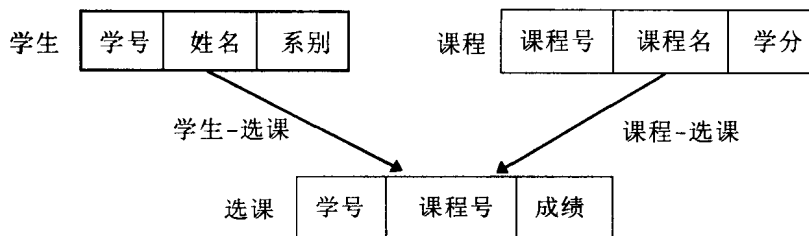


图 1.4 学生-选课-课程的网状模型

1.2.3 关系模型

在关系模型中，数据的逻辑结构是一张二维表，在数据库中，满足下列条件的二维表称为关系模型：

- (1) 每一列中的分量是类型相同的数据；
- (2) 列的顺序可以是任意的；
- (3) 行的顺序可以是任意的；
- (4) 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；
- (5) 表中的任意两行不能完全相同。

表 1.1 给出的工资表便是一个关系模型。

表 1.1 工资表

编号	姓名	部门名称	基本工资	补贴	奖金	房租	水电	应发	实发
01	李红	汽车系	400.00	20.00	200.00	20.00	10.00	620.00	590.00
02	王伟	汽车系	600.00	50.00	200.00	50.00	15.00	850.00	785.00
03	沈华	公路系	700.00	55.00	200.00	50.00	10.00	955.00	895.00
04	王艳	公路系	300.00	20.00	200.00	20.00	10.00	520.00	490.00
05	任美华	管理系	450.00	30.00	200.00	30.00	10.00	680.00	640.00

关系数据库采用关系模型作为数据的组织方式。

层次数据库是数据库系统的先驱，而网状数据库则为数据库在概念、方法、技术上的发展奠定了基础。它们是数据库技术中研究最早的两种数据库，而且也曾得到广泛的应用，但是这两种 DBMS 存在着结构比较复杂、用户不易掌握、操作比较复杂、数据存取操作必须按照模型结构中已定义好的存取路径才能进行等缺点，这就限制了这两种数据库管理系统的发展。

关系数据库以其具有严格的数学理论、使用简单灵活、数据独立性强等特点，而被公认为是最有前途的一种数据库管理系统，发展十分迅速，已成为占主导地位的 DBMS。自 20 世纪 80 年代以来，作为商品推出的 DBMS 几乎都是关系型的，如 System R、INGRES、ORACLE、SYBASE、INFORMIX、dBASE III、dBASE IV、FoxBASE+、FoxPro 等。

1.3 关系数据库

1.3.1 基本概念

前面已经介绍了关系数据库的基本数据结构是满足一定条件的二维表。为讨论关系数据库,先给出关系模型中的一些基本概念。

关系(Relation): 一个二维表就叫做一个关系。

属性(Attribute): 表中的一列叫做一个属性,有时也叫做一个字段(Field)

域(Domain): 一个属性的取值范围叫做一个域。

元组(Tuple): 表中的一行叫做一个元组,有时也叫做一条记录(Record)。

码(Key): 表中的某个属性,若它的值唯一地标识一个元组,则称该属性为“候选码”。若一个关系有多个候选码,则选定其中一个为主码,这个属性称为主属性。

分量: 元组中的一个属性值叫做元组的一个分量。

关系模式: 是对关系的描述,它包括关系名、组成该关系的属性名、属性到域的映像。通常简记为:

关系名(属性名₁, 属性名₂, ..., 属性名_n)

属性到域的映像通常直接说明为属性的类型、长度等。

关系数据库: 采用关系模式作为数据的组织方式的数据库叫做关系数据库。对关系数据库的描述,称为关系数据库的型,它包括若干域的定义以及在这些域上定义的若干关系模式。这些关系模式在某一时刻对应的关系的集合,称为关系数据库的值。

例如,表 1.2 中的关系是一个人事档案表。

表 1.2 人事档案表

编号	姓名	性别	婚否	出生日期	部门名称	职称	工资
0001	李红	女	未婚	70.12.02	汽车系	讲师	400.00
0002	王伟	男	已婚	65.02.03	汽车系	教授	600.00
0003	沈华	女	已婚	54.01.15	公路系	教授	700.00
0004	王艳	女	未婚	72.10.14	公路系	助教	300.00
0005	任华美	女	已婚	65.08.19	管理系	讲师	450.00
0006	赵同刚	男	未婚	74.10.12	管理系	助教	350.00
0007	周飞	男	已婚	54.06.03	电子系	教授	750.00
0008	张倩	女	已婚	65.12.25	电子系	讲师	450.00

表中每一行是一个职员的记录,是关系的一个元组,编号、姓名、性别、出生日期等均是属性。其中编号是唯一识别一条记录的属性,因此称为“主码”。对于编号这一属性,域是“0001”~“9999”,对于姓名属性,域是2~4个汉字组成的字符串,对于性别属性,域是{男,女}。

人事档案表的关系模式可记为:

人事档案表(编号, 姓名, 性别, 婚否, 出生日期, 部门名称, 职称, 工资)

一个关系模式在某一时刻的内容(称为相应模式的状态),是元组的集合,称为关系。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。

1.3.2 关系操作

关系操作可以用两种方式表示:关系代数与关系演算。已经证明,这两种方式是等价的。这里仅以关系代数来表示关系操作。

常用的关系操作有两类:传统的集合操作和专门的关系操作。

传统的集合操作,如集合并、交、差、广义笛卡尔积。这类操作将关系看成元组的集合,其操作是从关系的“水平”方向,也就是二维表的行的角度来进行的。

专门的关系操作,如选择、投影、连接等。这类操作不仅涉及关系的水平方向(即二维表的行),而且涉及关系的竖直方向(即二维表的列)。

关系代数的操作对象是关系,操作的结果仍为关系。

这里对于大家所熟悉的集合操作不再介绍,仅介绍专门的关系操作。

(1) 选择(Selection)

选择操作即在关系中选择满足某些条件的元组。也就是说,选择操作是在二维表中选择满足指定条件的行。例如,在人事档案表中找出所有女职工的元组,就可以用选择操作实现,条件是:性别等于“女”。

(2) 投影(Projection)

投影操作是在关系中选择某些属性列。例如,要在人事档案表中找出职称为教授的所有职工的编号、姓名、工资,则可以用投影操作实现,选择表中职称为教授的职工的编号、姓名、工资列。操作结果见表 1.3。

表 1.3

编号	姓名	工资
0002	王伟	600.00
0003	沈华	700.00
0007	周飞	750.00

(3) 连接(Join)

连接操作是从两个关系的笛卡尔积中选取属性间满足一定条件的元组。例如,表 1.4 中的关系 R 和表 1.5 中的关系 S, R 和 S 的广义笛卡尔积为表 1.6 中的关系 T。

表 1.4 关系 R

编号	姓名	系号	性别
0001	李红	1	女
0002	王伟	1	男
0007	周飞	3	男

表 1.5 关系 S

系号	系名称
1	汽车系
2	公路系
3	管理系
4	电子系

做条件为“R.系号=S.系号”的连接操作,所得结果为关系 U,是关系 T 的一个子集,如表 1.7 所示。