

BUSINESS ADMINISTRATION

21世纪工商管理系列教材

《数据库程序设计教程 (Visual FoxPro 6.0) (第二版)》

学习指导书

周山芙 编著

 中国人民大学出版社

21 世纪工商管理系列教材

**《数据库程序设计教程
(Visual FoxPro 6.0)(第二版)》
学习指导书**

周山芙 编著

中国人民大学出版社

图书在版编目 (CIP) 数据

《数据库程序设计教程 (Visual FoxPro 6.0) (第二版)》学习指导书/周山芙编著.
北京: 中国人民大学出版社, 2007
(21 世纪工商管理系列教材)
ISBN 978-7-300-08772-6

I. 数…

II. 周…

III. 关系数据库-数据库管理系统, Visual FoxPro 6.0-高等学校-教学参考资料

IV. TP311.138

中国版本图书馆 CIP 数据核字 (2007) 第 190489 号

21 世纪工商管理系列教材

《数据库程序设计教程 (Visual FoxPro 6.0) (第二版)》学习指导书

周山芙 编著

出版发行	中国人民大学出版社	邮政编码	100080
社 址	北京中关村大街 31 号	010-62511398 (质管部)	
电 话	010-62511242 (总编室)	010-62514148 (门市部)	
	010-82501766 (邮购部)	010-62515275 (盗版举报)	
	010-62515195 (发行公司)		
网 址	http://www.crup.com.cn		
	http://www.ttrnet.com(人大教研网)		
经 销	新华书店	版 次	2007 年 12 月第 1 版
印 刷	北京市鑫霸印务有限公司	印 次	2007 年 12 月第 1 次印刷
规 格	170mm×228mm 16 开本	定 价	29.80 元
印 张	20.00		
字 数	364 000		

版权所有 侵权必究 印装差错 负责调换

本书是教材《数据库程序设计教程 (Visual FoxPro 6.0) (第二版)》的配套学习指导书。考虑到一般学校的授课时间都比较紧张,主教材只涉及了 Visual FoxPro 6.0 的基本内容;而学习指导书实际上是主教材内容的进阶和补充,同时提供了习题的答案及很多复杂的上机练习题详解。

本书是教材《数据库程序设计教程 (Visual FoxPro 6.0) (第二版)》的配套学习指导书。考虑到一般学校的授课时间都比较紧张,主教材只涉及了 Visual FoxPro 6.0 的基本内容;而学习指导书实际上是主教材内容的进阶和补充,同时提供了习题的答案及很多复杂的上机练习题详解。

前言

本书是教材《数据库程序设计教程 (Visual FoxPro 6.0) (第二版)》的配套学习指导书。考虑到一般学校的授课时间都比较紧张,主教材只涉及了 Visual FoxPro 6.0 的基本内容;而学习指导书实际上是主教材内容的进阶和补充,同时提供了习题的答案及很多复杂的上机练习题详解。

学习指导书的结构与主教材完全一致,每个章节都一一对应。指导书的章节基本上是按照“内容的补充与提升”和“习题及答案”两大块设计的,最后一章增加了“系统开发案例”的介绍。

本书仍然力图坚持深入浅出、由浅入深、循序渐进的风格,并且着重介绍上机操作的方式方法;本书从辅助教材的角度出发,没有再重新完整介绍每个章节的内容,而是结合主教材的内容进行了补充和提升。这样的安排一方面可以满足感兴趣读者进一步学习的要求,另一方面也为授课教师备课提供了方便;特别是对于准备自学 Visual FoxPro 的读者,或是需要参加全国计算机二级数据库考试的读者,再或是希望利用 Visual FoxPro 解决自己工作中一些实际问题的科技、管理人员,本学习指导书将是非常实用和必要的。

对于实际例题或案例本书仍然介绍 Visual FoxPro 的可视化操作方法,同时介绍了许多复杂程序的设计方法、设计技巧以及 SQL 语言的解决方法。读者可以按照书中的例题上机练习,也可以到出版社相关网站上查找这些案例程序。

周山芙
于中国人民大学明德楼

目 录

第1章	Visual FoxPro 概述	1
1.1	内容的补充与提升	1
1.2	习题及答案	23
第2章	数据库的创建与操作	32
2.1	内容的补充与提升	32
2.2	习题及答案	41
第3章	程序设计基础	48
3.1	内容的补充与提升	48
3.2	习题及答案	83
第4章	基本对象的程序设计	105
4.1	内容的补充与提升	105
4.2	习题及答案	135

5 章		
	输入输出界面的程序设计	153
	5.1 内容的补充与提升	153
	5.2 习题及答案	190
6 章		
	查询处理界面设计	206
	6.1 内容的补充与提升	206
	6.2 习题及答案	218
7 章		
	接口界面设计	228
	7.1 内容的补充与提升	228
	7.2 习题及答案	232
8 章		
	帮助界面的程序设计	255
	8.1 内容的补充与提升	255
	8.2 习题及答案	264
9 章		
	创建应用程序系统	287
	9.1 内容的补充与提升	287
	9.2 习题及答案	295
	9.3 系统开发案例	297
	参考文献	311

1

章

Visual FoxPro 概述

1.1 内容的补充与提升

1.1.1 数据库的基本术语

1. 数据和信息

信息至今尚未有一个统一、确切的定义。不同的研究领域对信息的内涵有不同的理解，于是形成了不同的定义和描述。

从广泛意义上说，信息是对事物运动状态和特征的描述，它反映的是关于某一客观系统中，某一事物的属性或表现形式。例如，气象是一个客观事物。同样，为了描述这个事物，人们可能说，明天是风和日暖的好天气。这些就是描述气象的信息，由此可以决定是否出游等。同样，为了在计算机中存储和处理这个信息，就必须把它的属性和特征抽象出来；当然人们会根据处理的需要每次只提取部分必要的特征。比如，温度、风向风力、相对湿度、降水概率、晴/雨，并且可以描述成：

气象：(12/26，南转北 1~2 级，60%，30%，晴转多云)

人们通过这些数据就可以判断出天气的好坏。这些记录信息的物理符号就是数据，这样一组数据的集合叫做一条记录。数据所反映的客观事物属性是它的内容，而表示数据内容的符号则是它的形式。

为了表示不同性质的内容,数据形式通常有三种:数值型数据,即对客观事物进行定量记录的符号,如体重、年龄、价格、温度、湿度等;字符型数据,即对客观事物进行定性记录的符号,如姓名、单位、地址、风向、晴/雨等;特殊型数据,如声音、视频、图形、图像等。

从计算机处理的角度看,数据泛指那些可以被计算机接受,并能够被计算机处理的(数字化)符号。在数据库中数据是存储和管理的基本对象。

同时,信息也定义为是人类认识了的数据,是数据的含义;通俗地讲,信息是经过加工处理并能对人类客观行为产生影响的有用数据的表现形式。比如,气象:(12/26,南转北1~2级,60%,30%,晴转多云)是一条天气好坏的信息,而12/26,南转北1~2级……都仅仅是数据。数据表示了信息,而信息必须以数据形式表现才能被人们理解。

2. 数据处理

所谓数据处理,实际上包括对各种类型数据进行加工的处理操作及把处理过的数据合理组织、存储,随时为用户服务的管理操作。数据处理包括对数据的采集、整理、存储、加工、传输等;数据管理包括分类、编码、组织、存储、检索、维护等。

数据处理的目的是不但要从大量的、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,而且要管理好这些数据信息,以便人们能够随时提取和使用它们。所以说数据处理是基础,数据管理是核心。

随着电子计算机软件和硬件技术的发展,数据处理过程发生了划时代的变革,而数据库技术的发展,又使数据处理跨入了一个崭新的阶段。数据处理技术的发展经历了三个阶段:

(1) 人工管理阶段。没有专门的软件对数据进行管理,在程序中既要考虑处理过程,又要考虑数据的定义和组织,程序和数据总是联系在一起。它主要适合早期主要应用于数据量很少的科学计算。其中程序和数据之间的对应关系如图1—1所示。

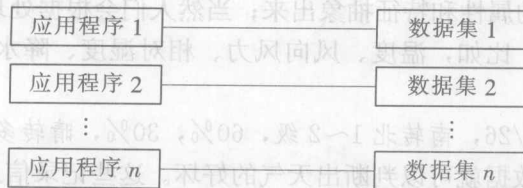


图 1—1 人工管理阶段

(2) 文件系统阶段。将数据从程序中分离出来，组成相互独立的数据文件。文件系统建立了数据文件内部的数据结构并可以管理所有数据文件，每个程序通过文件系统和自己相应的数据文件联系。但是数据文件之间没有联系，或者说文件系统在整体上是无结构的，它适用于科学计算及简单数据的管理。其中程序和数据之间的对应关系如图 1—2 所示。

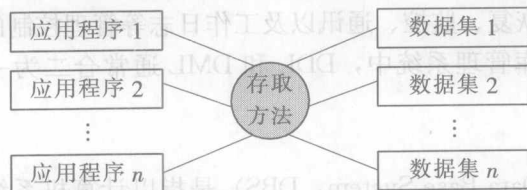


图 1—2 文件系统阶段

(3) 数据库系统阶段。一方面实现了数据与程序的完全独立，另一方面又实现了数据的统一管理。众多程序或应用需要的各种数据，全部交给数据库系统管理，大大压缩了冗余数据，实现了多用户、多应用的数据的共享。其中程序和数据之间的对应关系如图 1—3 所示。

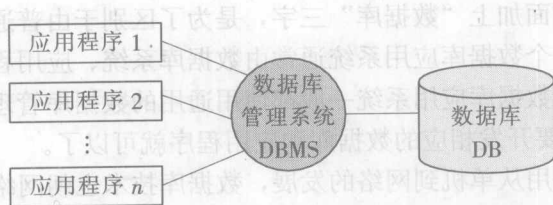


图 1—3 数据库系统阶段

3. 数据库

数据库 (Data Base, DB) 是以一定的组织方式将相关数据组织在一起并存储在计算机外存储器上所形成的、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。

数据库不是根据某个用户的需要，而是按照信息的自然联系构造数据；它能以最佳的方式、最少的冗余，为多个用户或多个应用共享服务的系统。

4. 数据库管理系统

数据库管理系统是用户建立、使用和管理数据库的软件系统，简称为 DBMS (Data Base Management System)。数据库管理系统是数据库系统的核心，DBMS 通常有下列三个基本部分组成：

(1) 数据描述语言 (Data Description Language, DDL)。用来描述数据库、表的结构,供用户建立数据库及表。

(2) 数据操作语言 (Data Manipulation Language, DML)。供用户对数据表进行数据的查询 (包括检索与统计) 和存储 (包括增加、删除与修改) 等操作。

(3) 其他管理和控制程序。实现数据库建立、运行和维护时的统一管理、统一控制,从而保证数据的安全、完整及多用户并发操作。同时完成初始数据的输入、转换、转存、恢复、监控、通讯以及工作日志等管理控制的实用程序。

在小型的数据库管理系统中,DDL 和 DML 通常合二为一,成为一体化的语言。

5. 数据库系统

数据库系统 (Data Base System, DBS) 是指以计算机系统为基础,以数据库技术管理大量共享数据的综合系统。它一般应当由数据库、计算机软硬件系统、数据库管理系统和用户 (最终用户、应用程序设计员和数据库管理员) 四个部分构成。人们习惯上常常把数据库系统简称为数据库,但是应当注意和前面说的数据库 (相关数据集合) 概念相区别。

一个具体的数据库系统也称为数据库应用系统 (Data Base Application System, DBAS)。

在应用系统前面加上“数据库”三字,是为了区别于由普通文件管理系统支持的应用系统。一个数据库应用系统通常由数据库系统、应用程序和实际数据构成。在微机上建立数据库应用系统一般都使用通用的数据库管理系统 (如 Visual FoxPro), 用户只要开发相应的数据库和应用程序就可以了。

随着计算机应用从单机到网络的发展,数据库技术也朝网络应用的方向有了新的发展。其中最主要的是从单用户应用扩展到多用户应用;从集中式应用扩展到分布式应用。早期的微机 DBMS 都是单用户系统,比如 FoxBASE 等;而 Visual FoxPro 是应用于微机网络上的多用户数据库管理系统。

1.1.2 数据库系统的特点

1. 全面的数据结构化

在数据文件系统中,数据文件内部是有结构的,但数据文件之间不存在联系,因此从数据的整体来说数据文件系统是没有结构的。数据库虽然也由许多独立的数据集合 (表文件) 组成,并且文件内部具有完整的数据结构,但是在数据库中各表文件之间也建立了完善的相互联系,所以说数据库系统具有全面的数据结构。这种特征特别能够反映现实世界的的数据联系,适应大批量数据管理的客观需要。

2. 数据共享

共享是数据库系统的目的,也是它的重要特点。在文件系统中,一批数据总是由特定用户专用的。而一个数据库中的数据,不仅可以为同一企业或组织的内部各部门使用,还可以为不同组织、地区,甚至不同国家的用户使用。这就是数据库系统的数据共享特点。

3. 可控冗余度

在文件系统中,由于每个应用都拥有并使用自己的数据,各数据文件中难免有许多数据相互重复,这就是数据冗余。数据库系统是面对整个系统共享数据而建立的,各个应用的数据集中存储,共同使用;这就自然不必存放重复数据,可以尽可能地避免数据的冗余。当然为了建立各表文件之间的联系,还需要保留一些相关数据,但是数据冗余度是可以控制的。

4. 统一的管理和控制

所有用户的数据都集中在数据库中,所以通过数据库管理系统软件包统一管理数据也是数据库系统的显著特点。由于数据库要为多用户提供共享数据,而用户对数据的存取往往又是并发进行的,所以数据库系统必须具有安全性控制、完整性控制、并发性控制和数据库数据恢复功能。

5. 数据独立性

在文件系统中,数据结构和应用程序是相互依赖的,任何一方的改变总会使另一方的改变。在数据库系统中,这种相互依赖性是很小的,数据和程序具有相对的独立性。数据库的建立独立于程序,是通过三级模式来描述的,如图 1-4 所示。

在数据库三级模式中,数据结构具有物理结构和逻辑结构两个方面。描述物理结构的称为存储模式,它直接与操作系统及硬件联系。一个数据库系统只能建立一个存储模式。

描述逻辑结构的称为概念模式(或模式),它是数据库数据的完整表示,是所有用户的公共数据视图。一个数据库系统只有一个模式,它总是以某一种逻辑数据模型为基础,统一考虑所有用户的要求,并有机地综合成一个逻辑整体。模式仅仅是数据逻辑模型的描述,不涉及具体数据值。模式的一个具体值称为模式的一个实例,一个模式往往有许多实例。模式是相对稳定的,而实例是不断变动的。因为模式反映的是数据库的结构,一旦定义好以后基本就不再变动;而实例反映的是数据库某个时刻的状态,数据库的数据是在不断更新变化的。

针对每一个用户或应用,又由模式导出子模式(或叫用户模式)。子模式是直接面向用户的,用户能够看见并使用的局部数据的逻辑结构描述。每一个子模

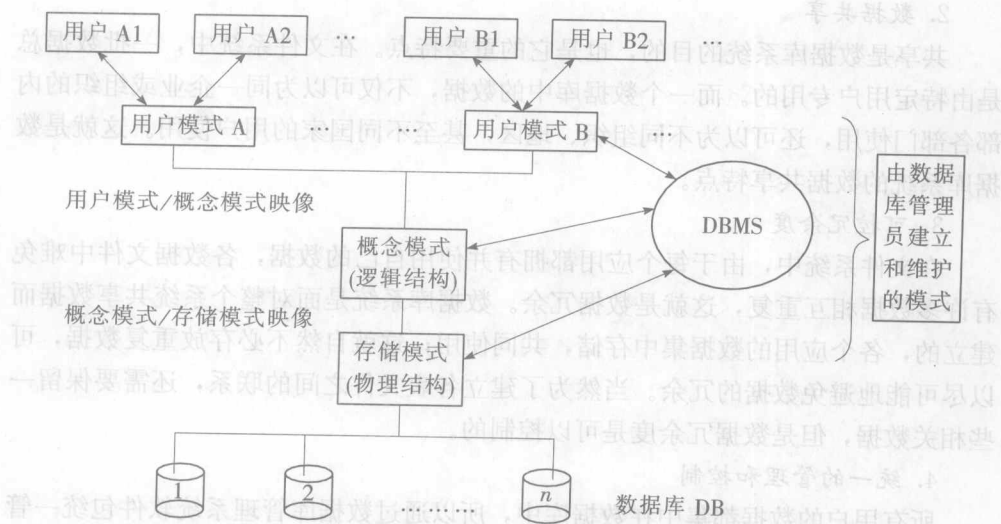


图 1—4 三级模式

式都是模式的一个子集；也可以把它看成是模式的一个窗口。一个数据库系统可以有多个子模式。

特别是，在三级模式中提供了两个映像功能：一个是在存储结构与逻辑结构之间的映像（转换）功能；另一个是在逻辑结构与用户结构之间的映像（转换）功能。第一种映像使得数据存储结构改变时逻辑结构不变，因而相应的程序也不变，这就是数据库的物理独立性；第二种映像使得逻辑结构改变时，用户结构不变，应用程序也不用改变，这是数据和程序的逻辑独立性。这种独立性，使得应用程序的编写再不用考虑数据的描述和存取问题，从而大大减少了应用程序的修改和维护工作。

1.1.3 数据模型

对现实世界事物特征的模拟和抽象就是这个事物的模型。人们对现实世界事物的研究，总是通过对它的模型研究实现的。特别是如果需要用计算机来研究处理现实世界的具体事物，必须先把具体事物转换为抽象的模型，再转换为计算机可以处理的数据模型，从而以模拟的方式实现对现实世界事物的处理。所以说，数据模型是抽象、表示和处理现实世界中事物的基本工具。

建立数据模型应当做到：一是真实反映现实世界；二是容易被人理解；三是便于在计算机上实现。能够真实地反映（模拟）现实世界是根本要求，但既要人

容易理解,又要计算机便于理解实现,就很不容易了。因此,建模的过程往往采用逐步抽象的方法,并把数据模型划分为两大类:以人的观点模拟现实世界的模型叫做概念模型或信息模型;以计算机系统的观点模拟现实世界的模型叫做数据模型。

概念模型和数据模型属于信息世界和机器世界两个层次,如图 1—5 所示。为了把现实世界中的具体事物抽象成计算机能够处理的数据和信息,必须首先按人的认识观点将现实世界的具体事物抽象为属于信息世界的概念模型,然后再将概念模型进一步转换为机器(计算机)世界的数据库模型。从图上可以看出,数据模型还应当细分为逻辑数据模型和物理数据模型。



图 1—5 模型层次的划分

1. 概念模型

概念模型是从现实世界到信息世界的抽象,是数据库设计人员设计数据库的工具,同时又是与用户交流的语言。概念模型按用户的观点对现实世界建模,是缺乏计算机知识的基本用户最容易理解的、便于和数据库设计人员进行交流的语言。它独立于任何数据库管理系统,但是又很容易向数据库管理系统支持的逻辑数据模型转换。描述概念模型最常用的工具就是实体——联系图(Entity-Relationship Approach),也叫 E-R 图。

描述概念模型的基本术语如下:

(1) 实体(Entity)。客观存在,并且可以互相区别的事物称为实体。它可以是具体的人、物,如李四、中国人民大学、桌子、某一本图书。也可以是抽象的概念,如交通法、ASCII 代码。还可以是两个实体之间的某种联系,如学生和课程实体之间的联系——选课,商店和顾客实体之间的联系——购物。

(2) 实体型(Entity Type)。用实体名描述它的各属性名,可以刻画出全部同质实体的共同特征和性质,它被称为实体型。比如,张三图书{书号、书名、作者、出版社、价格}就是一个实体型。

(3) 实体集(Entity Set)。可以包含在某个实体型中的全部实体,叫做实体集。比如,某人目前拥有的全部图书,就是一个图书实体集。例如:

ISDN1001	计算机基础	李四	中国人民大学出版社	27.00
ISDN2103	高等数学	李名	清华大学出版社	20.00
...	...	...	...	...

(4) 属性 (Attribute)。实体具有的每一个特性都称为一个属性。选择的实体属性越多, 刻画出的实体越清晰。属性有“型”和“值”的区分, 属性的名称 (说明) 就是属性的“型”; 对型的具体赋值就是属性的“值”。比如图书可以由书号、书名、作者、出版社、价格等属性型的序列来描述。而属性值 (9012-076, 英语, 陈琳, 人民教育出版社, 20.00) 的集合则描述了一本具体的图书。

(5) 码 (Key)。在众多属性中能够唯一标识 (确定) 该实体的属性或属性组称为实体的码。比如, “书号”就可以是图书实体的码。

(6) 域 (Domain)。属性的取值范围称为该属性的域。比如, 图书的价格属性域可以是 1~20。

(7) 联系 (Relationship)。一个实体内部各属性之间是互相关联的, 叫做实体内部联系。在各实体集之间也存在着关联, 称为实体的外部联系。这些联系可以分为一对一、一对多和多对多三类。

- 一对一联系。当前实体集中的每一个实体, 在另一个实体集中最多只能找到一个可以与它相对应的实体; 反之, 在另一个实体集中的每一个实体, 也只能在当前实体集中最多找到一个能够相对应的实体。那么这两个实体集之间就存在着一对一的联系, 并记做 1:1。

比如, 企业实体和总经理实体之间的联系, 就是一对一的联系。因为一个企业只能有一位总经理, 而一个总经理又只能领导一个企业。用图形表示如图 1—6 (a) 所示。

再比如在学生实体中, 每个学生的学号属性与姓名 (假定没有同名同姓) 属性之间存在着一对一联系。这是实体内部的联系, 可以用图 1—6 (b) 表示。

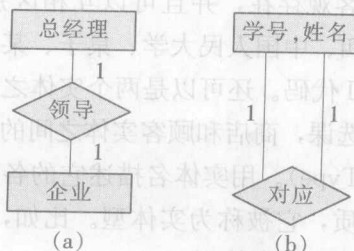


图 1—6 一对一联系

- 一对多联系。当前实体集中的每一个实体, 在另一个实体集中可以找到多

一个能够与它相对应的实体；反之，在另一个实体集中的每一个实体，却只能在当前实体集中找到一个能够相对应的实体。那么这两个实体集之间就存在着一对多的联系，并记做 $1:n$ 。

比如，每个教师一般都能开出多门课程，若每门课程只能由一个教师开设，那么“教师”和“课程”的联系，就是一对多的联系。如果反过来看“课程”和“教师”的联系就是多对一的联系了。用图形表示如图 1—7 (a) 所示。

再比如，在教师实体中，每一种职称（如教授）都会有多位老师，所以职称属性与职工编号属性之间就是内部一对多的联系，可以用图 1—7 (b) 表示。

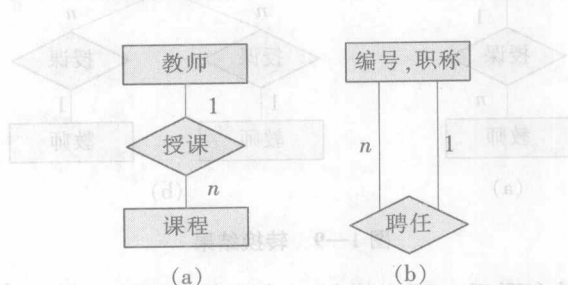


图 1—7 一对多联系

● 多对多联系。当前实体集中的每一个实体，在另一个实体集中可以找到多个能够与它相对应的实体；反之，在另一个实体集中的每一个实体，也能够当前实体集中找到多个能够相对应的实体。那么这两个实体集之间就存在着多对多的联系，并记做 $m:n$ 。

比如，每个教师都能开出多门课程，而每门课程又允许由多个教师开设，那么“教师”和“课程”的联系，就是多对多的联系，如图 1—8 (a) 所示。

再比如，在教师实体中，如果有同名同姓的教师存在，因为每种职称可以聘任多位教师，而同一个人又可能是不同的职称，所以姓名属性与职称属性之间就存在着多对多的联系。这种对应关系如图 1—8 (b) 所示。

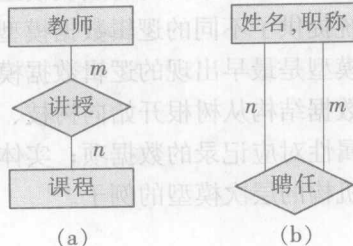


图 1—8 多对多联系

实际上一对一联系是一对多联系的特例, 一对多联系又是多对多联系的特例。在实际处理中常常将两个具有一对一联系的实体合并成一个实体; 而将一个具有多对多联系的实体分解为两个具有一对多联系的实体。比如, 对于具有内部多对多联系的教师实体, 可将“职称”信息从教师实体中抽出来单独建立一个实体集, 将所有职称实体都集中在这里, 这样, “职称”实体和“教师”实体之间就成了一对多联系, 如图 1—9 (a) 所示。

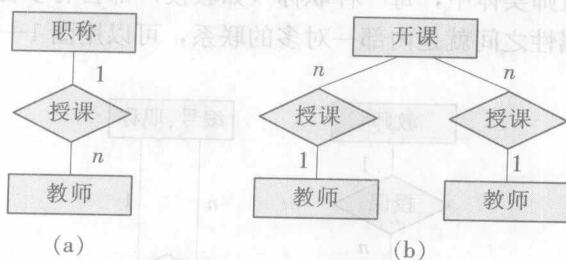


图 1—9 转换结果

对于外部多对多联系, 可以增加一个中间实体, 使得一个多对多联系变成两个一对多联系。比如, 创建一个“开课”实体, 包含了学校所有老师开设的全部课程记录。那么“教师”实体与“开课”实体之间, 以及“课程”实体与“开课”实体之间分别存在着一对多联系。如图 1—9 (b) 所示。在这个转换的过程中, “开课”实体起到了一种中间纽带的作用, 所以就把它叫做纽带实体。

2. 逻辑数据模型

逻辑数据模型是用户通过计算机系统中数据库管理系统看到的现实世界, 它描述了数据库数据整体结构。逻辑数据模型是数据库系统的核心和基础, 它通常由数据结构、数据操作和数据完整性约束三部分概念组成。数据结构是对系统静态特性的描述, 它是逻辑数据模型中最重要的部分, 所以人们一般以逻辑数据结构的类型来命名该数据模型。其中常见的有层次模型、网状模型、关系模型三类。不同的数据库管理系统提供了不同的逻辑数据模型。

(1) 层次模型。层次模型是最早出现的逻辑数据模型。层次模型以树型结构来表示实体之间的联系, 数据结构从树根开始向树枝、树叶逐层展开。实体用记录 (类型) 表示, 实体的属性对应记录的数据项; 实体之间的联系用有向连线表示。图 1—10 是学校组织机构的层次模型的例子。