



普通高等教育“十二五”规划教材
高等学校公共课**计算机**规划教材



数据库应用基础

——基于Visual FoxPro 9.0

(第2版)

■ 王 衍 主编



COMPUTER
TECHNOLOGY



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

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

省精品课程教材

普通高等教育“十二五”规划教材

高等学校公共课计算机规划教材

数据库应用基础

——基于Visual Fox Pro 9.0（第2版）

王 衍 主编

金 勤 林 锋 赵 辉 陈明晶 编著

电子工业出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

本书是浙江省精品课程教材,以 Visual FoxPro 9.0 版本为环境,介绍数据库的基本知识、数据库及数据表的操作与管理,并较为全面地介绍面向过程的程序设计方法与面向对象的程序设计方法,为运用数据库系统构建一个管理信息系统打下基础。全书共 9 章,主要内容包括:数据库基础知识,数据表的基本操作,数据库的建立与操作,结构化程序设计,面向对象程序设计基础,常用表单控件的使用,表单设计应用,查询、视图及报表设计,应用程序的管理及编译等,每章后附习题和实验。本书配套《数据库应用基础学习指导——基于 Visual FoxPro 9.0 (第 2 版)》,并提供免费电子课件和习题参考答案。

本书可作为高等学校非计算机专业数据库及其程序设计应用的基础教材,也可供从事数据库系统教学、研究和应用的广大师生和工程技术人员学习参考。

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

数据库应用基础:基于 Visual FoxPro 9.0 / 王衍主编. —2 版. —北京:电子工业出版社,2012.1

高等学校公共课计算机规划教材

ISBN 978-7-121-15393-8

I. ①数… II. ①王… III. ①关系数据库—数据库管理系统, Visual FoxPro 9.0—高等学校—教材

IV. ④TP311.138

中国版本图书馆 CIP 数据核字(2011)第 252558 号

策划编辑:王羽佳

责任编辑:王羽佳

印刷:北京市顺义兴华印刷厂

装订:三河市双峰印刷装订有限公司

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:565 千字

印 次:2012 年 1 月第 1 次印刷

印 数:6 000 册 定价:39.90 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 zlts@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

本书是浙江省精品课程教材。Visual FoxPro 作为一个关系数据库管理系统软件,从诞生起就一直是高等学校非计算机专业,特别是经济管理类专业选用的计算机教学语言之一。20余年来,从 dBASE、FoxBASE 到 Visual FoxPro 之所以一直长盛不衰,不仅是因为 Visual FoxPro 本身作为微软的产品在 Windows 平台上不断发展完善,更主要的是,这一系统软件集程序设计和数据库语言于一体。在程序设计方面,既支持传统的面向过程程序设计,又支持目前广泛采用的面向对象程序设计;在数据库方面,既有自身的特点,又支持 SQL—SELECT 标准的数据库结构查询语言,同时,Visual FoxPro 还支持 Web 服务,具有跨平台数据应用能力。因此,作为继“大学计算机基础”的后续课程,选择 Visual FoxPro 作为“数据库应用基础”或“程序设计基础”课程的环境进行教学是比较恰当的。

本书根据教育部提出的非计算机专业计算机基础课程教学要求编写,主要具有以下特点。

1. 以 Visual FoxPro 9.0 为基础,反映最新成果。尽管作为基础教学,主要应用的是 Visual FoxPro 的基本内容,但 Visual FoxPro 9.0 的新增功能同样会给学习基础的学生带来许多方便,也为学生更进一步深入学习提供了更高的平台。

2. 突出应用,强化实验。教无止境,参加本书编写的团队由长期从事计算机基础教学的一线教师组成,经过反复实践,不断总结提升,积累了宝贵的经验。经管类非计算机专业学习计算机程序设计和数据库的目的是提升学生的科技素养,使他们具备在信息系统构建中与信息技术专业人员沟通的能力。用什么方法来有效提高学生的这种能力呢?作者认为,只有在讲清基本概念的基础上,通过大量实例的讲解和实验,才能增强学生运用程序和数据库解决问题的能力。本书提供了丰富的习题,精心设计了 17 个实验,与教学内容同步配套。

3. 体系完整,内容简洁清晰。本书在保持全书内容体系完整的同时进行了取舍,突出了基础性和应用性。理论概念的叙述方式、章节顺序的安排、例题讲解的形式、习题和实验的设计等各个环节,均根据教学实际情况进行了仔细考虑。

本书共分 9 章,主要内容包括:数据库基础知识,数据表的基本操作,数据库的建立与操作,结构化程序设计,面向对象程序设计基础,常用表单控件的使用,表单设计应用,查询、视图及报表设计,应用程序的管理及编译。本书配套《数据库应用基础学习指导——基于 Visual FoxPro 9.0 (第 2 版)》,并提供免费电子课件和习题参考答案,请登录华信教育资源网(<http://www.hxedu.com.cn>)注册下载。

本书可作为高等学校非计算机专业数据库及其程序设计应用的基础教材,也可供从事数据库系统教学、研究和应用的广大教师、学生和工程技术人员学习、参考。

在教学中,可以考虑安排 51~68 学时的理论教学和 34 学时的实验教学,其中,第 8、9 章的内容可根据学生掌握情况进行取舍。

本书由王衍主编并统稿。第 1、5 章由王衍编写,第 2、6 章由金勤编写,第 3、9 章由陈明晶编写,第 4 章由林锋、金勤编写,第 7、8 章由赵辉编写。

本书的编写参考了近年来出版的相关技术资料，吸取了许多专家和同仁的宝贵经验。同时，本书在编写过程中得到了浙江财经学院各级领导和同事的关心，以及浙江财经学院信息学院众多同事的全力支持，特别是在实验环节的设计上，是众多教师共同努力的结晶，在此向他们及所有关心和支持本书编写的老师表示真诚的感谢！

由于作者水平有限，书中难免有错误或不当之处，敬请读者批评、指正。

作 者

目 录

第 1 章 数据库基础	(1)	2.2 表内容的输入	(51)
1.1 数据库系统概述	(1)	2.2.1 表文件的打开与关闭	(51)
1.1.1 数据管理技术的发展	(1)	2.2.2 表记录的输入	(52)
1.1.2 数据库系统	(2)	2.2.3 表记录的显示	(55)
1.1.3 概念数据模型	(3)	2.3 表内容的编辑修改	(58)
1.1.4 逻辑数据模型	(4)	2.3.1 表记录的定位	(58)
1.2 关系模型	(5)	2.3.2 表记录的修改	(60)
1.2.1 关系的基本概念及关系 数据库	(5)	2.3.3 表记录的删除	(62)
1.2.2 关系的特征	(7)	2.4 表的排序与索引	(65)
1.2.3 关系操作	(8)	2.4.1 表记录的排序	(66)
1.2.4 数据库的完整性规则	(8)	2.4.2 索引的概念与索引文件 类型	(67)
1.3 Visual FoxPro 概述	(9)	2.4.3 索引的建立	(69)
1.3.1 Visual FoxPro 的发展及 特点	(9)	2.4.4 索引文件的打开、关闭及主控 设置	(71)
1.3.2 Visual FoxPro 集成开发 环境	(11)	2.4.5 表内容检索	(75)
1.4 Visual FoxPro 语言基础	(13)	2.5 表的其他操作	(77)
1.4.1 数据类型	(13)	2.5.1 表内容的统计	(77)
1.4.2 数据存储	(16)	2.5.2 表内容的复制与大批量 数据追加	(81)
1.4.3 表达式	(21)	2.5.3 表结构文件的建立与 应用	(83)
1.4.4 常用函数	(25)	2.6 常用的文件管理命令	(84)
1.4.5 Visual FoxPro 命令格式与 文件类型	(36)	2.6.1 列文件目录命令	(84)
1.5 小结	(38)	2.6.2 显示文件内容命令	(85)
习题 1	(38)	2.6.3 复制磁盘文件命令	(85)
实验 1 Visual FoxPro 环境与表达式、 常用函数的使用	(42)	2.6.4 更改磁盘文件名命令	(85)
第 2 章 数据表的基本操作	(46)	2.6.5 删除磁盘文件名命令	(86)
2.1 创建数据表	(46)	2.7 小结	(86)
2.1.1 表的概念	(46)	习题 2	(87)
2.1.2 创建表结构	(46)	实验 2.1 数据表结构的建立 与记录输入	(89)
2.1.3 表结构的显示与修改	(50)	实验 2.2 数据表记录的定位、删除 与索引	(91)

实验 2.3 数据表记录的查找、统计 及文件操作	(94)	4.2.2 选择结构	(135)
第 3 章 数据库的建立与操作	(96)	4.2.3 循环结构	(139)
3.1 数据库的建立	(96)	4.2.4 多种结构的嵌套	(144)
3.1.1 建立数据库	(96)	4.3 子程序和用户自定义 函数	(149)
3.1.2 数据库表和自由表的相互 转换	(98)	4.3.1 子程序	(149)
3.1.3 数据库的基本操作	(99)	4.3.2 过程和过程文件	(151)
3.1.4 工作区	(101)	4.3.3 变量的参数传递及 作用域	(152)
3.1.5 建立表间的临时关联	(106)	4.3.4 自定义函数	(155)
3.2 数据库的完整性实现	(108)	4.4 小结	(158)
3.2.1 长表名和表注释	(109)	习题 4	(158)
3.2.2 长字段名和字段属性	(110)	实验 4.1 顺序与选择程序 设计	(165)
3.2.3 设置记录规则	(111)	实验 4.2 循环结构程序 设计	(166)
3.2.4 主索引与表间的永久 关联	(112)	实验 4.3 循环嵌套结构程序 设计	(167)
3.2.5 参照完整性实现	(114)	实验 4.4 过程文件与自定义 函数	(169)
3.3 控制共享数据的访问	(115)	第 5 章 面向对象程序设计 基础	(170)
3.3.1 数据表的打开方式	(115)	5.1 面向对象程序设计简介	(170)
3.3.2 锁定数据	(116)	5.1.1 面向对象程序设计示例	(170)
3.3.3 数据解锁	(118)	5.1.2 对象与类	(172)
3.4 结构化查询语言 SQL	(119)	5.1.3 对象的属性、事件与 方法	(173)
3.4.1 SELECT 数据查询 语句	(119)	5.2 表单文件的创建与对象的 添加	(175)
3.4.2 SQL 语言的其他常用 语句	(122)	5.2.1 表单文件的创建与运行	(175)
3.5 小结	(123)	5.2.2 在表单中添加对象	(177)
习题 3	(123)	5.3 对象引用、属性设置及事件的 编辑与响应	(181)
实验 3 Visual FoxPro 数据库 设计	(126)	5.3.1 对象的引用	(181)
第 4 章 结构化程序设计	(129)	5.3.2 对象属性的设置	(182)
4.1 程序文件设计	(129)	5.3.3 事件代码的编辑、响应及调用 顺序	(184)
4.1.1 结构化程序设计基础	(129)	5.3.4 在表单中添加属性和 方法	(188)
4.1.2 程序设计的基本命令	(130)		
4.1.3 程序文件的建立与 执行	(132)		
4.2 程序控制的基本结构	(133)		
4.2.1 顺序结构	(133)		

5.4 小结	(188)	7.2 交互对话界面设计	(246)
习题 5	(189)	7.3 与数据源有关的表单 设计	(247)
实验 5 面向对象程序设计 入门	(191)	7.4 多表表单设计	(250)
第 6 章 常用表单控件的 使用	(194)	7.5 SQL 查询设计实例	(252)
6.1 输出类控件	(194)	7.6 表单集的设计	(256)
6.1.1 标签	(194)	7.7 小结	(258)
6.1.2 图像、线条与形状	(195)	习题 7	(258)
6.2 输入类控件	(196)	实验 7 多表表单的设计与 应用	(261)
6.2.1 文本框	(196)	第 8 章 查询、视图及报表 设计	(263)
6.2.2 编辑框	(198)	8.1 查询设计	(263)
6.2.3 列表框与组合框	(200)	8.1.1 查询设计器	(263)
6.2.4 微调	(205)	8.1.2 用 SQL 语言实现查询 设计	(266)
6.3 控制类控件	(206)	8.2 视图设计	(267)
6.3.1 命令按钮与命令 按钮组	(206)	8.2.1 视图设计器	(267)
6.3.2 复选框与选项按钮组	(210)	8.2.2 视图的 SQL 语句	(270)
6.3.3 计时器	(214)	8.3 创建报表	(271)
6.4 容器类控件	(216)	8.3.1 使用报表向导创建报表	(271)
6.4.1 表格	(216)	8.3.2 使用报表器创建快速 报表	(273)
6.4.2 页框	(220)	8.4 利用报表设计器设计 报表	(274)
6.4.3 容器	(222)	8.4.1 报表设计器的组成和数据 环境	(274)
6.5 链接与嵌入类控件	(224)	8.4.2 报表布局与报表带区	(275)
6.5.1 ActiveX 控件	(224)	8.4.3 设置报表控件	(276)
6.5.2 ActiveX 绑定控件	(228)	8.4.4 常用的报表控件操作	(278)
6.5.3 超级链接	(229)	8.4.5 报表设计举例	(278)
6.6 小结	(229)	8.5 小结	(280)
习题 6	(230)	习题 8	(281)
实验 6.1 利用输出类控件设计 表单	(236)	实验 8 查询、视图及报表 设计	(283)
实验 6.2 利用输入类控件设计 表单	(237)	第 9 章 应用程序的管理及 编译	(285)
实验 6.3 利用控制类控件设计 表单	(239)	9.1 菜单设计	(285)
实验 6.4 利用容器类控件及 OLE 控件设计表单	(242)		
第 7 章 表单设计应用	(244)		
7.1 信息显示界面设计	(244)		

9.1.1 菜单设计基本步骤·····	(285)	9.3.2 应用程序编译·····	(297)
9.1.2 菜单设计工具·····	(288)	9.3.3 应用程序发布·····	(297)
9.1.3 快捷菜单设计·····	(291)	9.4 小结·····	(301)
9.1.4 在应用程序中使用菜单···	(292)	习题 9·····	(301)
9.2 主文件设计·····	(293)	实验 9 一个简单的营销管理信息 系统设计·····	(302)
9.2.1 主文件·····	(293)	参考文献·····	(304)
9.2.2 错误处理程序设计·····	(294)		
9.3 编译应用程序·····	(296)		
9.3.1 管理项目管理器中的 文件·····	(296)		

第 1 章 数据库基础

当今世界，无论人们身在何处，都离不开计算机信息系统的支持，而在任何一个信息系统的背后，又都需要数据库系统的支撑。数据库是一门研究数据管理的技术，在信息时代，学习和掌握一些数据库方面的知识是十分有意义的。

本章内容主要包括：数据库技术的发展、数据库系统的构成、概念模型、数据模型，着重阐述关系模型及关系模型下的数据完整性规则，介绍 Visual FoxPro 的基本功能、特点及语言基础。

1.1 数据库系统概述

人类的社会活动离不开数据处理，所谓数据处理包括对数据的收集、存储、加工、分类、排序、检索、传播等一系列工作。处理数据的目的是为了管理好数据，使之成为对决策有用的信息。数据库技术就是针对数据管理的计算机学科的一个重要分支，并随着计算机技术的发展而逐渐发展和完善起来。

1.1.1 数据管理技术的发展

数据管理技术与计算机硬件、软件及计算机应用的发展有着密切的联系，主要经历了人工管理、文件系统和数据库系统三个阶段。

1. 人工管理阶段

20 世纪 50 年代中期以前，计算机主要用于科学计算。当时计算机的外存只有纸带、卡片、磁带，没有磁盘等直接存取的存储设备，并且缺少必要的操作系统及数据管理软件的支持。所以，这一阶段数据管理的特点如下。

① 数据不具有独立性，数据与程序不可分割，当数据结构发生变化后，对应的程序也必须做相应的修改。

② 数据不能长期保存，只是在需要计算某一题目时将数据输入，处理完成后就释放。

③ 没有专门的数据管理软件，数据的存储结构、存取方式、输入/输出方式均由程序员设计完成。

人工管理阶段应用程序与数据之间的对应关系可用图 1-1 表示。

2. 文件系统阶段

20 世纪 50 年代后期至 60 年代后期，计算机不仅用于科学计算，而且还大量用于信息管理。这时，硬件上已有了硬盘、磁鼓等直接存取设备。软件方面，操作系统中已经有了专门的文件系统来管理外存中的数据文件。这一阶段数据管理的特点如下。

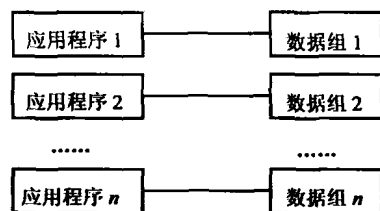


图 1-1 人工管理阶段应用程序与数据间的关系

① 程序和数据分开存储，数据可以以文件的形式独立地存放在外存中。

② 出现了专门的软件（文件系统）对数据文件进行存取、修改、插入和删除等操作管理，

程序员不必关心数据在存储器上存储的物理细节以及与外存交换的过程。

③ 文件系统仍然存在着这样一些问题：数据没有完全独立，文件系统上的文件还是为某一应用程序服务的；由于数据文件相互独立，数据文件之间缺乏联系，造成了数据冗余度大；由于相同数据的重复存储，数据不能集中管理，给数据的修改、维护带来了困难，容易造成数据的不一致性。

文件系统阶段应用程序与数据之间的对应关系如图 1-2 所示。

3. 数据库系统阶段

20 世纪 60 年代后期以来，计算机用于信息管理的规模和领域更加庞大，数据量急剧增加，数据共享和集中管理的需求越来越强烈，从而推动了数据库技术的发展。硬件方面有了大量的外存储器，软件方面研制了专门的数据库管理系统。数据库系统阶段主要特点如下。

① 面向整个系统组织数据，实现数据共享，允许多个应用程序和多个用户存取数据库中的数据。

② 减少了数据的冗余度，既减少了存储空间和存取时间，又可避免数据之间的不相容性和不一致性。

③ 具有较高的数据和程序的独立性，包括物理独立性和逻辑独立性两个方面。其中，物理独立性是指当数据的存储结构改变时，数据的逻辑结构可以不改变，从而程序也不必改变；而逻辑独立性则是指当数据的总体逻辑结构改变时，可以保持局部逻辑结构不变，程序员根据局部逻辑结构编写的应用程序也无须改变。

④ 有统一的数据控制功能，有较高的数据安全性、完整性，实现并发控制。

⑤ 提供数据排序、统计、分析、制表等多种数据操作。

数据库系统阶段应用程序与数据之间的对应关系可用图 1-3 表示。

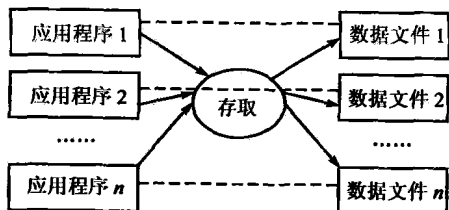


图 1-2 文件系统阶段应用程序与数据间的关系

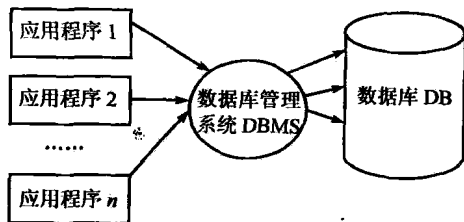


图 1-3 数据库系统阶段应用程序与数据间的关系

1.1.2 数据库系统

数据库系统是一个存储介质、处理对象和管理系统的集合。数据库系统通常由计算机硬件及相关软件、数据库、数据库管理系统及用户四部分组成。

1. 数据库

数据库（DB，Data Base）是人们为解决特定的任务，按一定的结构和组织方式存储在外存储器中的相关数据的集合。它具有最小的数据冗余，可供多个用户共享，独立于具体的应用程序。

2. 数据库管理系统

数据库管理系统（DBMS，Data Base Management System）是在操作系统支持下工作的操纵和管理数据的系统软件，是整个数据库系统的核心。它对数据库进行统一的管理和控制，以保证数据库的安全性和完整性。用户通过 DBMS 访问数据库中的数据，数据库管理员也通过 DBMS 进行数据库的维护工作。它提供多种功能，可使多个应用程序和用户用不同的方法

建立、修改和查询数据库，并提供对数据的排序、统计、分析、制表等功能。

3. 计算机硬件及相关软件

数据库系统是建立在计算机系统之上的。在硬件方面，它需要基本的计算机硬件（主机和外设）支持；在软件方面，需要操作系统（Windows、UNIX、Linux 等）、各种宿主语言（Visual C++、Visual Basic 等）和一些数据库辅助应用程序等。

4. 用户

数据库系统的用户通常有 3 种：一是对数据库系统进行日常维护的数据库管理员，二是用数据操纵语言和高级语言编制应用程序的程序员，三是使用数据库中数据的终端用户。

1.1.3 概念数据模型

数据库设计的过程是，根据人们要解决的问题，首先分析出与问题有关的实体及其属性，再分析出各实体之间的关系（概念数据模型），然后按照数据库管理系统所能支持的数据模型形成数据库（逻辑数据模型）。这里先讨论概念数据模型。

信息世界是客观事物（现实世界）在人脑中的反映，客观事物在信息世界中称为实体，反映实体之间联系的模型称为实体模型或概念模型。

1. 实体及实体间的联系

客观事物之间都存在着联系，这是由事物本身的性质所决定的。例如，在学校的教学管理系统中有教师、学生和课程，教师为学生授课，学生选修课程并取得成绩；在企业的销售系统中有职工和商品，职工销售商品，并取得销售业绩；等等。

（1）实体（Entity）

客观存在并且可以相互区别的事物称为实体。实体通常指可以触及的具体事物，如一个职工、一名学生、一件商品、一本书等；实体也可以指抽象事件，如一次销售、一次借阅或一场足球比赛。

（2）属性（Attribute）

实体具有的特性称为属性。例如，学生的学号、姓名、年龄，商品的名称、类别、单价等。属性用类型（Type）和值（Value）来表征，每个属性都有值域（Domain）。

（3）联系（Relationship）

在现实世界中，事物内部及事物之间是有联系的，这些联系在信息世界中反映为实体内部的联系和实体之间的联系。实体内部的联系通常是指组成实体的各属性之间的联系。

（4）实体集（Entity Set）

性质相同的同类实体的集合称为实体集，例如所有职工、全体学生等。

2. 实体联系的类型

根据联系的实际情况，可以将实体间的联系归纳为 3 种类型。

（1）一对一联系（1:1）

如果实体集 E1 中的每一个实体只能与实体集 E2 中的一个实体有联系，反之亦然，则称实体集 E1 与实体集 E2 是一一对一的联系，表示为 1:1。例如，一个职工对应一张照片，一张照片一定是某个职工的。

（2）一对多联系（1:n）

如果实体集 E1 中的每一个实体能与实体集 E2 中若干个实体有联系，而实体集 E2 中每一个实体至多与实体集 E1 中的一个实体有联系，则称实体集 E1 与实体集 E2 是一对多的联系，

表示为 $1:n$ 。例如，一名学生对应多门课程成绩，一个成绩只能对应某一名学生；一个职工可能有多笔销售业务，而一笔销售业务一定属于某一个职工。

通常，将一对多关系中的实体集 E1 称为“父”方或“一”方，实体集 E2 称为“子”方或“多”方。一对多关系是关系数据库中最常见的联系类型。

(3) 多对多联系 ($m:n$)

如果实体集 E1 中的每一个实体能与实体集 E2 中若干个实体有联系，而实体集 E2 中的每一个实体也能与实体集 E1 中若干个实体有联系，则称实体集 E1 与实体集 E2 是多对多的联系，表示为 $m:n$ 。例如，一名学生可以选若干门课程学习，而每门课程也可以有多名学生选择；一个职工可以销售多个商品，而每一个商品也可以有多个职工销售。

3. 概念模型的表示方法

概念模型是对信息世界的建模，表示方法很多，其中最为常用的是实体-联系方法 (Entity-Relationship Approach)。该方法用 E-R 图来描述概念模型。E-R 图的基本组成元素有实体、属性和联系等。

① 实体：用矩形框表示，实体名称写在框内。

② 属性：用椭圆形表示，框内注明属性的名称，属性与实体之间用实线连接。

③ 联系：用菱形框表示实体间的相互关系，框内注明联系的名称。应当注意的是，联系本身也是一种实体，也可以有属性。

④ 连线：用无向连线来连接实体与属性、实体与联系、联系与属性，在进行实体与联系连接时应标明对应关系，即 $1:1$ ， $1:n$ ， $m:n$ 。图 1-4 给出了职工、销售、商品实体集之间的 3 种联系的 E-R 图。

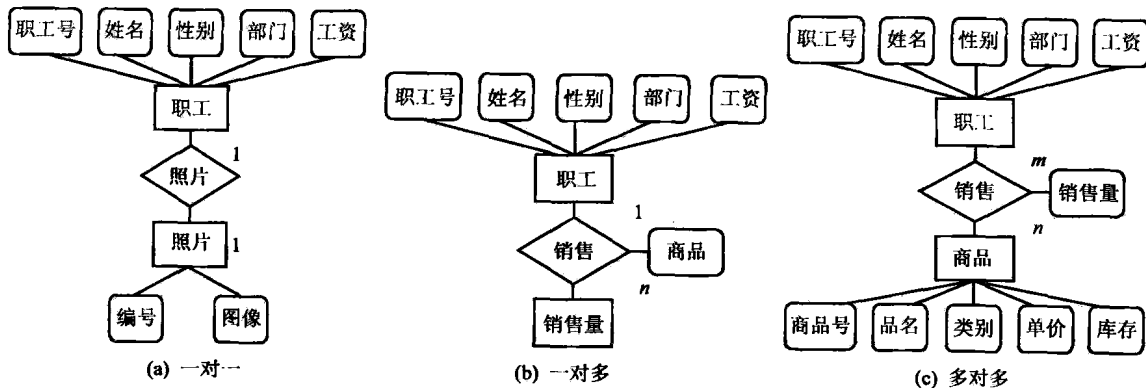


图 1-4 实体集间的 3 种联系

1.1.4 逻辑数据模型

数据库中的数据是按照一定的结构和组织方式存放的，这种结构用数据模型来表示。数据模型（也称逻辑数据模型）是现实世界数据特征的抽象。设计一种数据模型要根据应用系统中所涉及的数据性质、内在联系、管理要求等来组织。目前，比较流行的数据模型有 3 种，即按图论理论建立的层次结构模型和网状结构模型，以及按关系理论建立的关系结构模型。数据库通常分为层次式数据库、网络式数据库和关系式数据库 3 种。而不同的数据库是按不同的数据结构来联系和组织数据的。

1. 层次模型

层次模型也称树状模型，实质上是一种有根结点的定向有序树，如图 1-5 所示，其模型结

构具有如下特点:

- ① 有且仅有一个结点无父结点, 称为根结点, 其层次最高;
- ② 一个父结点向下可以有多个子节点, 一个子结点向上有且仅有一个父结点。

树根与枝点之间的联系称为边, 树根与边之比为 1:N, 即树根只有一个, 树枝有 N 个。各结点之间是一种“一对一”或“一对多”的关系。

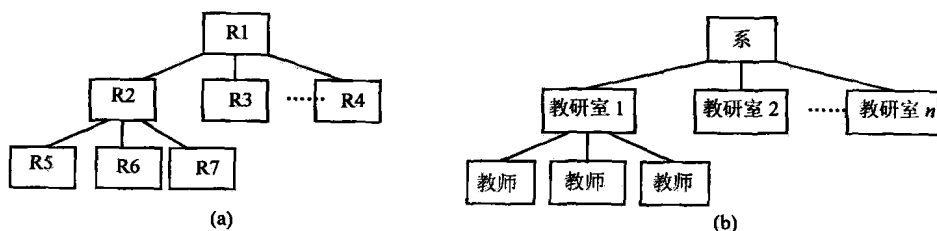


图 1-5 层次模型示意图

2. 网状模型

广义地讲, 任意一个连通的基本层次联系的集合为网状模型, 如图 1-6 所示, 其特点如下:

- ① 可以有一个以上的结点无父结点 (如 R1、R2、R4);
- ② 至少有一个结点有多于一个以上的父结点。

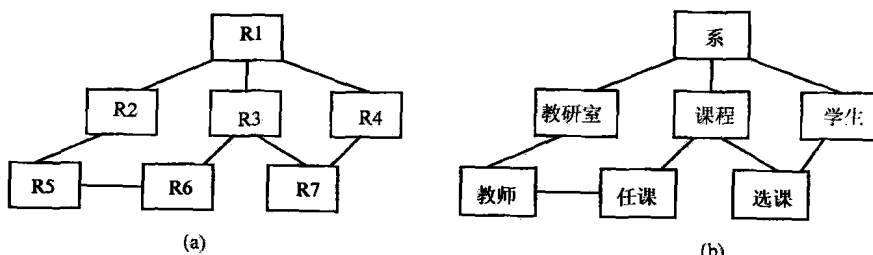


图 1-6 网状模型示意图

3. 关系模型

关系模型是目前最流行的一种逻辑数据模型。1970 年, IBM 公司的研究员 E.F.Cold 首次提出了数据库系统的关系模型, 为关系模型数据库技术奠定了理论基础。由于他的杰出贡献, 1981 年, E.F.Cold 荣获计算机界最负盛名的 ACM 图灵奖。

关系模型是以二维表格来表示实体集中实体之间的联系的。关系模型中, 一张二维表称为一个关系, 并给它赋予一个名称, 称为关系名。关于关系模型数据库的基本知识将在 1.2 节中进行较详细的介绍。

1.2 关系模型

20 世纪 80 年代以来, 计算机厂商推出的数据库管理系统几乎都支持关系模型。目前, 比较流行的以关系模型为基础的大型数据库管理系统包括 Oracle、Sybase、Informix 和 SQL Server 等, 小型数据库管理系统则包括 Visual FoxPro、Access、Paradox 和 Betrive 等。

1.2.1 关系的基本概念及关系数据库

1. 关系的基本概念

(1) 关系

一个关系就是一张二维表, 每个关系都有一个关系名。例如, 图 1-7 所示的职工情况表就

是一个二元关系，该表格清晰地反映出单位职工的基本情况。如果表名为“职工”，则“职工”即为关系名。在 Visual FoxPro 中，一个关系对应一个数据文件，关系名对应于文件名。

关系名：职工 或 文件名：职工.dbf

关系模式 或表结构	职工号	姓名	性别	婚否	出生日期	基本工资	部门	简历	照片	属性或字段
元组或 记录	199701	李长江	男	T	05/12/75	2500.00	直销	memo	gen	属性值或 字段值
	199702	张伟	男	F	06/23/76	2300.00	零售	memo	gen	
	199801	李四方	男	T	06/18/77	2000.00	零售	memo	gen	
	199803	赵英	女	T	03/19/75	2600.00	客服	memo	gen	
	199804	洪秀珍	女	T	12/25/76	2100.00	直销	memo	gen	
	200001	张军	男	T	05/11/77	2200.00	零售	memo	gen	
	200005	孙学华	女	F	02/17/75	2300.00	客服	memo	gen	
	200006	陈文	男	T	08/08/74	2000.00	直销	memo	gen	
	200601	张丽英	女	F	04/23/82	1500.00	零售	memo	gen	
	200602	王强	男	F	10/23/83	1500.00	直销	memo	gen	

图 1-7 职工情况表

(2) 元组

二维表中的每一行在关系中称为元组。一个元组对应于数据文件中的一条记录，所以元组也称记录。例如，图 1-7 中姓名为“李长江”的所在行的所有数据就是一个元组，称为一条记录。

(3) 属性

二维表中的每一列在关系中称为属性，每个属性都有一个属性名，属性的取值范围取决于各个元组的属性值。例如，图 1-7 中的第 2 列属性，“姓名”是它的属性名，“李长江”是其中的一个属性值。

在 Visual FoxPro 中，属性也就是数据文件中的字段，属性名就是字段名，而属性值对应于各个记录的字段值。

(4) 域

域是指二维表中属性的取值范围，取值范围也就决定了表中字段的宽度。例如，图 1-7 中属性“婚否”的取值范围是 T 或 F。

(5) 关系模式

对关系的描述称为关系模式，一个关系模式对应一个表的结构。表示为：

关系名 (属性 1, 属性 2, 属性 3, ……, 属性 n)

例如：

职工 (职工号, 姓名, 性别, 婚否, 出生日期, 基本工资, 部门, 简历, 照片)

销售 (职工号, 商品号, 数量)

商品 (商品号, 商品名称, 类别, 库存量, 单价, 单位)

(6) 关键字 (关键字段或码)

在关系中，可以用来唯一地标识一个元组的属性或属性组合称为关键字。单个属性的关键字称为单关键字，多个属性组合的关键字称为组合关键字。

例如，图 1-7 中的“职工号”是单关键字，因为每个职工的工号不允许重复。在销售 (职工号, 商品号, 数量) 关系中，它的关键字则是“职工号+商品号”的组合，因为一个职工可以销售多种商品，一个商品可以被多个职工销售，只有某个职工销售的某个商品则是确定的。

(7) 主关键字 (主键) 与候选关键字

通常在一个关系中，关键字可能不止一个，但主关键字只能有一个，并且取值是确定的。当某关键字被选作表的主关键字后，如果还有其他的关键字，则其他的关键字称为候选关键字。

例如，在“职工”表中，如果不存在同名同姓的职工记录，则姓名也可以是关键字，但如果选择“职工号”作为主关键字，则“姓名”就是候选关键字。

(8) 外部关键字

如果一个属性在本表中不是主关键字，而在另一个表中是主关键字，则该属性称为外部关键字。

例如，在“销售”表中的“职工号”是“职工”表的主关键字，但它并不是“销售”表的主关键字。这里，称“职工号”属性为“销售”表的外部关键字。

这里提醒大家注意：关系、元组及属性等是数学领域的术语，二维表、行和列等是日常用语，而表文件、记录和字段等则是计算机领域的术语，这些术语间是相互对应的。在 Visual FoxPro 中经常使用的是表文件、记录和字段等概念。

2. 关系表之间的联系

前面已经讨论了实体间的联系类型有 1:1、1:n 和 m:n 的联系，这种联系类型在关系数据库中就是表与表间的联系。不同的联系方式，在设计数据库表时应当考虑以下问题。

(1) 表与表之间是 1:1 的联系

对于 1:1 联系类型的信息，设计时可分为两个表或合并为一个表。如果要将一个表分成许多字段，或因安全原因而隔离表中的部分数据，则可以在表与表之间建立一对一的联系关系。例如，可以将“职工号”与“照片”的有关信息放在一个表中。

(2) 表与表之间是 1:n 的联系

对于 1:n 联系类型的信息，设计时需将父表中的主关键字放入子表中，以实现两表之间的有效关联。例如，“职工”表与“销售”表之间为一对多的联系。具体设计表时，需在“销售”表中加进“职工”表的主关键字“职工号”，作为其外部关键字。

(3) 表与表之间是 m:n 的联系

对于 m:n 联系类型的信息，设计时需要另外增加一个新表，这个表称为“关联表”，其中必须包括两个表的主关键字，再加入两表之间的关联字段。例如，“职工”表中的职工与“商品”表中的商品是多对多的联系，需要增加“销售”表，并在其中包括“职工号”和“商品号”，另外加上“销售数量”，作为“职工”表与“商品”表的关联字段。

3. 关系数据库

用关系模型建立数据和描述数据之间联系的数据库就是关系数据库。在关系数据库中，一个数据文件存放一个关系的数据，若干个相关数据文件的集合就组成了关系数据库。

在 Visual FoxPro 中，数据库由 4 个层次组成：字段、记录、表和数据库。一个关系数据库由若干个数据表组成，每个数据表又由若干条记录组成，而每条记录又是由若干字段组成的。属于某一数据库的表称为数据库表，而不属于任何数据库的表称为自由表。

本书所使用的职工营销数据库，主要由职工情况、销售情况和商品情况 3 张数据表组成，详见图 1-7~图 1-9。

职工号	商品号	数量
199701	1001	80.00
199702	1001	30.00
199803	2003	15.00
199701	2001	30.00
199804	3001	50.00
200001	2002	46.00
199801	3003	32.00
199803	1003	23.00
200601	1002	16.00
199702	2002	18.00

图 1-8 销售情况表

商品号	商品名称	类别	库存量	单价	单位
1001	海飞丝	洗涤	700.00	50.00	瓶
1002	潘婷	洗涤	580.00	40.00	瓶
1003	沙宣	洗涤	360.00	47.00	瓶
1004	飘柔	洗涤	400.00	38.00	瓶
2001	可口可乐	饮料	500.00	72.00	箱
2002	非常可乐	饮料	300.00	68.00	箱
2003	娃哈哈矿泉水	饮料	600.00	43.00	箱
3001	德芙巧克力	糖果	800.00	80.00	包
3002	大白兔奶糖	糖果	500.00	55.00	包
3003	话梅奶糖	糖果	430.00	38.00	包

图 1-9 商品情况表

1.2.2 关系的特征

在关系模型中，对关系有一定的要求，关系必须具有以下特征：

① 关系中的每一属性（字段）都是不能再分的基本数据项，即表中不允许有子表；

- ② 关系中同一属性（字段）必须具有相同的数据类型（如字符型或数值型等）；
- ③ 同一关系中不能出现相同的属性名（字段名）；
- ④ 关系中不应有内容完全相同的元组（记录）；
- ⑤ 关系中记录和字段的顺序可以任意排列，不影响关系表中所表示的信息含义。

1.2.3 关系操作

当对关系数据库进行查询等操作时，常常需要对关系进行一定的运算处理，其中，选择、投影和连接是关系的3种基本运算。

1. 选择

按照一定条件在给定关系中选取若干元组（即选取若干记录）的操作称为选择。选择的条件通过逻辑表达式给出，条件为真的记录被选出。例如，在图 1-7 所示的关系中，如果按照“婚否=F.”的条件进行选择操作，则可得到图 1-10 所示的结果。

职工号	姓名	性别	婚否	出生日期	基本工资	部门	简历	照片
199702	张伟	男	F	06/23/76	2300.00	零售	hcmogen	
200005	孙宝华	男	F	02/17/75	2300.00	客服	hcmogen	
200601	张丽英	女	F	04/23/82	1500.00	零售	hcmogen	
200602	王强	男	F	10/23/83	1500.00	直销	hcmogen	

图 1-10 选择操作结果

2. 投影

在给定关系中选取确定的若干属性（即选取若干字段）组成新的关系称为投影。例如，在图 1-7 所示的关系中，如果选取职工号、姓名和性别 3 个字段的数据输出，则可得到图 1-11 所示的结果。

3. 连接

连接是指将两个或两个以上关系模式通过公共的属性名（字段名）拼接成一个新的关系模式，生成的新关系中包含满足连接条件的元组（记录）。例如，根据职工、销售、商品三个关系模式中共有的职工号、商品号连接成一个新的关系模式，结果如图 1-12 所示。

职工号	姓名	性别
199701	李红	女
199702	张伟	男
199801	李四方	男
199803	赵英	女
199804	洪秀珍	女
200001	张军	男
200005	孙宝华	男
200006	陈文	男
200601	张丽英	女
200602	王强	男

图 1-11 投影操作结果

职工号	姓名	商品号	商品名称	数量
199701	李红	1001	海飞丝	80.00
199702	张伟	1001	海飞丝	30.00
199803	赵英	2003	娃哈哈矿泉水	15.00
199701	李红	2001	可口可乐	30.00
199804	洪秀珍	3001	德芙巧克力	50.00
200001	张军	2002	非常可乐	46.00
199801	李四方	3003	话梅奶糖	32.00
199803	赵英	1003	沙宣	23.00
200601	张丽英	1002	潘婷	16.00
199702	张伟	2002	非常可乐	18.00

图 1-12 连接操作结果

1.2.4 数据库的完整性规则

数据库系统在运行过程中，由于数据输入、编辑、非法访问、程序错误等各方面的原因，容易产生数据错误和混乱，为保证关系中数据的正确、有效，需要建立数据完整性的约束机制来加以控制。关系模型的完整性规则就是对关系的某种约束条件。数据库表之所以与自由表有区别，主要是因为数据库表可以受完整性规则的约束。

所谓数据库的完整性，是指数据库中数据的正确性、有效性和一致性。这是数据库系统应遵守的一项标准，它保证了数据库中数据的完整、可靠。

要实现数据库的完整性，必须在数据库的建立和操作过程中，遵循一定的完整性规则，并随时检查是否满足完整性规则约束。数据库的完整性规则分为实体完整性、参照完整性和