



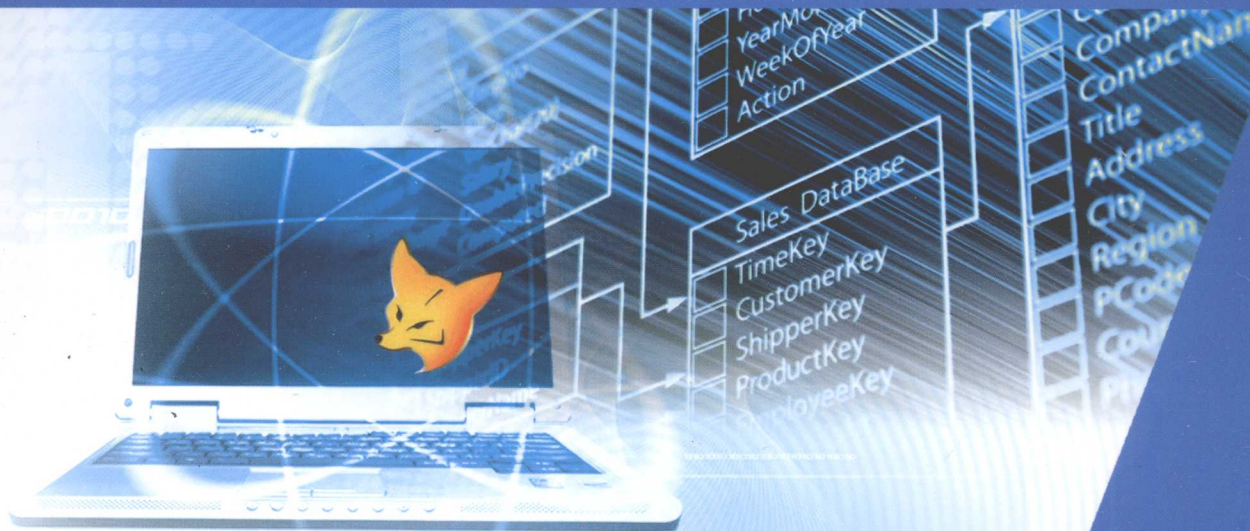
高等学校精品规划教材

Visual FoxPro 程序设计

(第二版)

主 编 王永国

副主编 黄晓梅 苗立刚 李治能



中国水利水电出版社
www.waterpub.com.cn

21 世纪高等学校精品规划教材

Visual FoxPro 程序设计

(第二版)

主 编 王永国

副主编 黄晓梅 苗立刚 李治能



中国水利水电出版社

www.waterpub.com.cn

内 容 提 要

本书是在总结编者多年教学实践、软件开发、试题库建设的基础上,考虑第一版使用读者的反馈信息,从实用、够用的角度出发,采用“案例驱动”的编写方式,通过190个实例介绍数据库系统基础知识及 Visual FoxPro 程序设计方法。全书共9章,内容包括数据库基础知识、Visual FoxPro 初步、Visual FoxPro 6.0 基础知识、数据库与数据表操作、查询与视图、结构化程序设计、表单设计、报表设计、菜单与工具栏设计、应用程序的开发。

为便于教学,每章前有内容导读、教学目标,每章后有小结和丰富的习题,以供读者练习与自测;同时编写了《Visual FoxPro 程序设计实训与考试指导(第二版)》(含光盘)作为配套辅导书,内容包括实训(含综合案例)、本书中所有习题的参考答案、Visual FoxPro 程序设计教学包软件介绍(内含教学课件、上机测试、无纸化练习系统等)。

本书可作为高等学校 Visual FoxPro 程序设计的教材,也可供从事计算机应用和开发的各类人员学习参考。另外,本书的编写兼顾了教育部考试中心指定的《全国计算机等级考试二级考试大纲(Visual FoxPro 程序设计)》的要求,因此也可作为计算机等级考试培训教材。

本书所配电子教案及教学相关资源可以从中国水利水电出版社网站和万水书苑下载,网址为 <http://www.waterpub.com.cn/softdown/>和 <http://www.wsbookshow.com>。使用本书的学校也可以与作者(ygwang21@163.com)联系,索取更多相关教学资源。

图书在版编目(CIP)数据

Visual FoxPro程序设计 / 王永国主编. -- 2版. --
北京:中国水利水电出版社,2009.12

21世纪高等学校精品规划教材
ISBN 978-7-5084-6993-5

I. ①V… II. ①王… III. ①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. ①TP311.138

中国版本图书馆CIP数据核字(2009)第212028号

策划编辑:雷顺加 责任编辑:杨元泓 加工编辑:陈洁 封面设计:李佳

书 名	21世纪高等学校精品规划教材 Visual FoxPro 程序设计(第二版)
作 者	主 编 王永国 副主编 黄晓梅 苗立刚 李治能
出版发行	中国水利水电出版社 (北京市海淀区玉渊潭南路1号D座 100038) 网址:www.waterpub.com.cn E-mail: mchannel@263.net(万水) sales@waterpub.com.cn 电话:(010) 68367658(营销中心)、82562819(万水)
经 售	全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	北京市天竺颖华印刷厂
规 格	184mm×260mm 16开本 19.5印张 477千字
版 次	2004年1月第1版 2009年12月第2版 2009年12月第5次印刷
印 数	20001—24000册
定 价	30.00元

凡购买我社图书,如有缺页、倒页、脱页的,本社营销中心负责调换
版权所有·侵权必究

再版前言

为了更好地服务于教学并兼顾相关考试,根据教育部《关于进一步加强高校计算机基础教学的意见》中有关“计算机程序设计基础课程教学基础要求”的精神,考虑到高等学校加强学生创新能力的培养方向,结合我们近年来从事“Visual FoxPro 程序设计”课程试题库建设的教学体会与开发经验,参考了第一版教材使用读者的反馈信息,我们编写了《Visual FoxPro 程序设计(第二版)》一书。

本书结构紧凑,教学内容设计合理,有利于教学,每一章都由内容导读、教学目标、任务导入、本章小结等部分组成。不仅将知识介绍和提高技能有机结合起来,而且具有很强的实用性。全书共分数据库基础知识和 Visual FoxPro 初步、Visual FoxPro 6.0 基础知识、数据库与数据表的操作、查询与视图、结构化程序设计、表单设计、报表设计、菜单与工具栏设计、应用程序的开发 9 章。采用“案例驱动”的编写方式,通过 190 个实例,从实用的角度出发,系统地介绍了数据库系统基础知识及 Visual FoxPro 程序设计方法,并通过“教学管理系统”应用程序开发案例,手把手介绍数据库应用程序开发方法和步骤,使你学后能具体地开发应用系统。

本书具有以下特点:

(1) 突出应用能力的培养,加强基础知识,紧跟计算机语言发展趋势,进一步加强了 SQL 语言的介绍,以适应当前广泛使用关系数据库系统的需要。

(2) 以 Visual FoxPro 6.0 简体中文版为依据,通过案例驱动教学,内容精炼、结构紧凑、通俗易懂、重点突出,注重实用与能力的培养,它克服了一般教科书学习 Visual FoxPro 枯燥、学了以后却不会用 Visual FoxPro 在开发环境中解决问题的难点,有助于提高学生的学习兴趣。

(3) 精选近 300 多道习题(选择、填空、阅读理解、操作、编程等),且都经过测试,并可在与本书配套的《Visual FoxPro 程序设计实训与考试指导(第二版)》一书中找到参考答案及其源代码,也可通过配套光盘在机器上练习评分。该配套辅导书由实训、习题解答、“Visual FoxPro 程序设计”课程教学包及其使用与考试指导 4 部分组成。为了适应新的教学需求、提高教学质量,配套光盘中含有我们开发的“Visual FoxPro 程序设计”课程教学包,该教学包可以说是一本学习 Visual FoxPro 的立体化教材,从 Visual FoxPro 电子课件、上机实训、综合案例,到试卷生成系统、Visual FoxPro 等级考试模拟训练系统等,信息量大,完全能够满足教师教学、学生自学测试的需要,对培养学生的实战能力与创新精神有重要指导作用。

本书由王永国任主编,黄晓梅、苗立刚、李治能任副主编。主要编写人员分工如下:王永国编写第 1、5 章,苗立刚编写第 3、4、6 章,黄晓梅编写第 7、8 章,王云编写第 2、9 章,李治能编写第 7、8 章。另外参与本书大纲讨论与编写、习题测试与校对的还有丁亚涛、王世好、郑尚志、尹荣章、宋万干、张仕江、王小云、谢倩、赵志艳、王多超、程中林等。

在本书出版过程中,参考了大量国内外教材、专著等材料,得到了许多同仁的关心和帮

助，中国水利水电出版社相关部门的领导和编辑也为本书的编辑和出版给予了大力支持和悉心指导，在此一并致谢。

由于作者才疏学浅，必有许多不足之处，望学术同仁不吝指教，以便我们再版时修正，如您需要下载课件、提供配套练习程序或案例、提出意见和建议，可通过 E-mail 与我们联系：ygywang21@163.com。

编者

2009年10月

目 录

再版前言	
第 1 章 数据库基础知识和 Visual FoxPro 初步 1	
1.1 数据库基础知识..... 1	
1.1.1 信息、数据、数据处理及数据管理 的发展..... 1	
1.1.2 数据库基本概念和术语..... 2	
1.1.3 数据库技术的发展..... 3	
1.1.4 数据模型..... 5	
1.2 关系数据库..... 6	
1.2.1 关系数据库..... 6	
1.2.2 关系运算..... 7	
1.2.3 关系的完整性约束..... 9	
1.3 Visual FoxPro 系统概述..... 9	
1.3.1 Visual FoxPro 的发展历史 与系统特点..... 9	
1.3.2 Visual FoxPro 的安装、启动与退出..... 10	
1.3.3 Visual FoxPro 的用户界面 与工作方式..... 11	
1.3.4 Visual FoxPro 集成开发环境..... 13	
1.4 项目管理器..... 20	
1.4.1 “项目管理器”与项目创建..... 20	
1.4.2 使用“项目管理器”..... 22	
1.4.3 定制“项目管理器”..... 23	
本章小结..... 25	
习题 1..... 25	
第 2 章 Visual FoxPro 6.0 基础知识 28	
2.1 Visual FoxPro 的命令格式与文件类型..... 28	
2.1.1 Visual FoxPro 的命令格式..... 28	
2.1.2 Visual FoxPro 的文件类型..... 30	
2.2 常用数据类型与基本数据元素..... 31	
2.2.1 数据类型..... 31	
2.2.2 常量与变量..... 32	
2.3 运算符与表达式..... 38	
2.3.1 运算符与表达式..... 38	
2.3.2 运算符的优先级..... 41	
2.4 常用函数..... 42	
2.4.1 数值函数..... 42	
2.4.2 字符函数..... 43	
2.4.3 日期和时间函数..... 44	
2.4.4 数据类型转换函数..... 45	
2.4.5 测试函数..... 47	
2.4.6 信息提示函数..... 48	
本章小结..... 49	
习题 2..... 50	
第 3 章 数据库与数据表的操作 53	
3.1 自由表的建立..... 53	
3.1.1 表的基本概念..... 54	
3.1.2 表结构的设计..... 54	
3.1.3 表结构的建立..... 55	
3.1.4 表数据的输入..... 57	
3.2 数据库的建立..... 58	
3.2.1 数据库表及自由表..... 58	
3.2.2 数据库的创建..... 59	
3.2.3 数据库的操作..... 61	
3.3 表的基本操作..... 71	
3.3.1 表结构的修改..... 71	
3.3.2 表的打开与关闭..... 73	
3.3.3 记录的显示..... 73	
3.3.4 记录的定位..... 76	
3.3.5 记录的增加..... 78	
3.3.6 记录的删除与恢复..... 79	
3.3.7 记录的修改..... 82	
3.3.8 表数据的导入与导出..... 85	
3.4 排序与索引..... 86	
3.4.1 排序..... 86	
3.4.2 索引..... 88	
3.5 查询与统计..... 95	

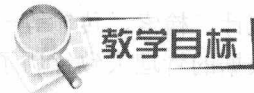
3.5.1 数据查询	95	5.3.2 格式化输入输出	156
3.5.2 数据统计	98	5.3.3 程序设计常用的其他命令	157
3.6 表之间的关系	101	5.4 程序流程控制	159
3.6.1 工作区	101	5.4.1 顺序结构	159
3.6.2 表间的几种关系	103	5.4.2 分支结构	159
本章小结	105	5.4.3 循环结构	163
习题 3	106	5.4.4 数组的应用	174
第 4 章 查询与视图	112	5.5 过程、自定义函数与参数传递	177
4.1 查询与视图的概念	112	5.5.1 过程和自定义函数	177
4.1.1 查询	112	5.5.2 变量作用域与参数传递	180
4.1.2 视图	113	5.6 程序的调试	183
4.1.3 查询与视图的区别	113	5.6.1 错误的类型	183
4.2 查询的建立	113	5.6.2 调试器的使用	183
4.2.1 利用向导建立查询	113	本章小结	186
4.2.2 使用查询设计器建立查询	116	习题 5	186
4.2.3 查询结果的输出	119	第 6 章 表单设计	197
4.3 视图的建立	120	6.1 面向对象的基本概念	197
4.3.1 利用向导建立视图	120	6.1.1 类和对象	197
4.3.2 利用视图设计器建立视图	123	6.1.2 属性、事件和方法	198
4.3.3 创建多表视图	126	6.1.3 基类和基类的类型	199
4.3.4 视图的修改与删除	126	6.1.4 自定义类	200
4.3.5 视图的作用	126	6.2 表单设计	201
4.4 关系数据库标准语言 SQL	127	6.2.1 创建、修改与运行表单	201
4.4.1 SQL 概述	127	6.2.2 表单属性和方法	204
4.4.2 SQL 的数据定义	128	6.2.3 表单设计器	207
4.4.3 SQL 的数据操纵	130	6.3 常用控件及其使用	215
4.4.4 SQL 的数据查询	132	6.3.1 标签 (Label) 控件	215
本章小结	144	6.3.2 命令按钮控件和命令按钮组	217
习题 4	144	6.3.3 文本框控件	220
第 5 章 结构化程序设计	149	6.3.4 编辑框控件	223
5.1 程序设计概述	149	6.3.5 单选按钮控件	223
5.1.1 程序	149	6.3.6 复选框 (Check) 控件	224
5.1.2 结构化和可视化程序设计	150	6.3.7 计时器 (Timer) 控件	225
5.2 程序文件的建立、修改与运行	150	6.3.8 图像控件	226
5.2.1 程序文件的建立	150	6.3.9 列表框与组合框控件	227
5.2.2 程序文件的修改	152	6.3.10 表格控件	232
5.2.3 程序文件的运行	153	6.3.11 页框控件	236
5.3 程序设计的常用命令	154	6.4 表单设计实例	237
5.3.1 基本输入输出	154	本章小结	240

习题 6	241	8.2.6 生成菜单	271
第 7 章 报表设计	246	8.2.7 运行菜单	272
7.1 报表向导	246	8.3 为顶层表单添加菜单和创建快捷 方式菜单	272
7.2 快速报表	251	8.3.1 为顶层表单添加菜单	272
7.3 报表设计器	252	8.3.2 创建快捷方式菜单	273
7.3.1 报表设计器的启动	253	8.4 创建自定义工具栏	274
7.3.2 报表设计器的数据环境	254	8.4.1 创建自定义工具栏的方法	274
7.3.3 “报表设计器”工具栏和 “报表控件”工具栏	255	8.4.2 在表单集中添加自定义工具栏	275
7.3.4 在报表中添加控件	256	8.4.3 创建自定义工具栏示例	275
7.3.5 数据分组	257	本章小结	277
7.3.6 报表的布局	258	习题 8	277
7.4 预览和打印报表	260	第 9 章 应用程序的开发	279
7.5 报表设计实例	261	9.1 数据库应用系统开发	279
本章小结	262	9.1.1 数据库应用系统开发	279
习题 7	263	9.1.2 数据库设计	281
第 8 章 菜单与工具栏设计	265	9.2 学生管理系统分析	283
8.1 菜单的规划与建立的基本步骤	265	9.2.1 学生管理系统的总体规划与设计	283
8.1.1 菜单的规划	265	9.2.2 数据库设计	284
8.1.2 菜单建立的基本步骤	266	9.3 学生管理系统的开发	285
8.2 建立菜单系统	266	9.3.1 学生管理系统的开发	285
8.2.1 创建快速菜单	266	9.3.2 制作发行盘	298
8.2.2 创建菜单项	268	本章小结	300
8.2.3 创建子菜单	269	习题 9	301
8.2.4 为菜单项分组	269	参考文献	303
8.2.5 为菜单项指定任务	269		

第 1 章 数据库基础知识和 Visual FoxPro 初步



本章从数据库系统所涉及的名词、术语及其相关概念出发，引导出数据库的发展、Visual FoxPro 的特点、启动及其集成化的用户界面，使读者从总体上对 Visual FoxPro 有一个基本认识，为后面进一步使用 Visual FoxPro 进行数据处理作必要的知识准备。



- 了解数据库管理技术的发展过程
- 掌握数据库技术的基本概念
- 掌握关系型数据库的概念，理解关系运算
- 掌握数据库的分类和特点
- 了解 Visual FoxPro 的安装、启动和退出
- 熟悉 Visual FoxPro 的工作界面，了解 Visual FoxPro 的工作方式
- 掌握 Visual FoxPro 项目管理器的使用方法

1.1 数据库基础知识



数据库管理系统是计算机科学的一个重要分支，它的产生和发展使数据处理进入了一个崭新的阶段。

如何将现实生活中的万事万物及其之间的关系转化成计算机可以识别的数据模型？如何将日常生活中广泛使用的二维表借助数据库进行处理？通过学习以下内容可以给出答案。

1.1.1 信息、数据、数据处理及数据管理的发展

1. 数据

在人们认识世界和改造世界的社会实践和日常生活中，为了进行相互交流，需要对周围的事实和发生的事件进行描述，描述的方式通常以抽象和形式化的方式进行，即抽取那些感兴趣的事物的特征或属性作为事物的描述。比如商店里出售的电视机，可描述成：长虹彩电，2976S 型，29 吋，……。这种对事实和事物描述的符号记录称为数据。数据可以是数

值、文字、图形、图像、声音等,它们都以二进制形式存储在计算机的存储器中。

2. 信息

信息泛指通过各种方式传播的,可以被人们感知的文字、声音、图形、图像、符号等所表征的某一特定事物的消息、知识或情报。人们根据信息决定自己的行动。通常,信息是人们通过解释、推论、归纳、分析、综合等方法对数据进行加工的结果,从这个意义上说,信息是以一种对人们有意义和有利的形式加工而成的数据。比如,通过对大量卫星云图数据进行分析计算后产生的天气预报数据,即为信息,人们据此进行决策(如出门是否带雨伞等)。



注意

数据与信息之间的关系可以表示为: 信息=数据+处理。

3. 数据处理

数据处理是对数据进行收集、组织、存储、加工、分类、维护、统计、检索、传输等一系列活动的总称。数据处理中,数据管理是其中一个重要的环节,对数据的管理大致经历了3个阶段:自由管理阶段、文件管理阶段和数据库管理阶段。

自由管理阶段的主要特点是程序依赖于数据,用户要与数据的物理地址打交道,编程难,效率低。

文件管理阶段是将数据组织成文件。该文件脱离程序而独立存在,并由文件管理系统通过文件名来管理数据。但是这种数据文件与程序紧密相关,它一旦离开了使用它的程序,其存在的价值便大打折扣,并且数据的冗余度大,一致性差,不能反映数据之间的关系。

为了克服上述缺陷,便产生了数据库管理方式。它与文件管理方式不同,其数据组织是面向整个系统,即用整体观点规划数据,形成一个数据中心,构成一个数据仓库,库中的数据能满足所有用户的不同要求,供不同用户所共享,应用程序不再与一个孤立的数据文件相对应,通过一个系统软件——数据库管理系统,实现逻辑文件与物理数据之间的映射。

1.1.2 数据库基本概念和术语

1. 数据库

数据库(DataBase, DB)是以一定的组织方式存储在一起的相关数据的集合,它能为多个用户共享,与应用程序彼此独立,并具有最小的冗余度。

2. 数据库管理系统

数据库管理系统(DataBase Management System, DBMS)是对DB进行管理的软件系统,Visual FoxPro便是一种数据库管理系统。DBMS的主要功能是对数据进行组织、存储、维护、处理和输出,具体包括以下功能:

(1) 数据库定义和数据装入功能,用户可使用它所提供的数据库定义语言(DDL)来创建数据库。

(2) 数据操纵功能。用它所提供的数据库操纵语言(DML),可实现对数据库的检索、插入、修改、删除等操作数据存取操作。

(3) 数据库的运行控制功能。这一部分是DBMS运行时的核心部分,包括安全性控制、数据完整性控制、多用户环境下的并发控制、系统故障后的数据库恢复以及数据库内部的维护(如索引、数据字典的自动维护等)。

(4) 数据库维护功能。其中包括数据库的重组功能和性能监视分析功能。

以数据库方式进行数据管理的好处如下:

(1) 数据的独立性强。它改变了数据从属于程序的作法,使数据和程序相互独立,库存中的数据能为多个程序共享,也能为多个用户所共享。

(2) 数据综合化、通用化,可实现共享。

(3) 数据的冗余度小。“冗余度”即相同数据在存储介质上重复出现的程度。由于数据独立于程序,因此相同的数据就不必多处存储。

(4) 具有统一的并发控制,安全性、完整性控制和数据库恢复功能。

3. 数据库系统

数据库系统(DataBase System, DBS)泛指由数据库、数据库管理系统、支持数据库运行的软、硬件环境以及数据库应用软件所构成的系统,其基本结构如图 1-1 所示。



图 1-1 数据库系统的基本结构

1.1.3 数据库技术的发展

1. 数据库技术萌芽

数据库技术萌芽于 20 世纪 60 年代中期,到 60 年代末 70 年代初出现了 3 个事件,标志着数据库技术日趋成熟,并有了坚实的理论基础。

(1) 1969 年 IBM 公司研制、开发了数据库管理系统商品化软件 IMS (Information Management System),IMS 的数据模型是层次结构的。

(2) 美国数据系统语言协会下属的数据库任务组 DBTG (DataBase Task Group)对数据库方法进行系统的讨论、研究,提出了若干报告,成为 DBTG 报告。DBTG 报告建立了数据库系统的许多概念、方法和技术。DBTG 所提议的方法是基于网状结构的,它是网状模型的基础和典型代表。

(3) 1970 年 IBM 公司 San Jose 研究实验室的研究员 E.F.Codd 发表了著名的“大型共享系统的关系数据库的关系模型”论文,为关系数据库技术奠定了理论基础。

2. 数据库技术的发展

自 20 世纪 70 年代开始,数据库技术有了很大的发展,基于关系数据模型的数据库技术结合了网络技术、面向对象等技术,产生了以下数据库系统:

(1) 对象—关系型数据库系统。随着计算机和网络技术的发展,数据库应用中出现了如 CAD (计算机辅助设计)/CAM (计算机辅助制造)、CIMS (计算机集成制造系统)、OA (办公自动化)、GIS (地理信息系统)、MIS (管理信息系统)、KBS (知识库系统)等新领域。这些领域的特点是:存储和处理的对象复杂,对象间的联系具有复杂的语义信息;需要复杂的数据类型支持、常驻内存的对象管理以及支持对大量对象的存取和计算;支持长事务和嵌套事务的处理。这些需求是传统关系数据库系统难以满足的。

(2) 分布式数据库。集中式数据库系统是将数据集中在一个数据库中。数据在逻辑上和物理上都是集中存放的。特点是访问方便,所有用户在存取和访问数据时,都要访问这个数据库。

分布式数据库系统 (Distributed DataBase System, DDBS) 是在集中式数据库基础上发展起来的, 是数据库技术与计算机网络技术、分布处理技术相结合的产物。分布式数据库系统是地理上分布在计算机网络不同结点, 逻辑上属于同一系统的数据库系统, 能支持全局应用, 同时存取两个或两个以上结点的数据。

(3) 面向对象数据库。面向对象数据库系统 (Object-Oriented DataBase System, OODBs) 是将面向对象的模型、方法和机制, 与先进的数据库技术有机地结合而形成的新型数据库系统。它从关系模型中脱离出来, 强调在数据库框架中发展类型、数据抽象、继承性和持久性; 它的基本设计思想是, 一方面把面向对象语言向数据库方向扩展, 使应用程序能够存取并处理对象, 另一方面扩展数据库系统, 使其具有面向对象的特征, 提供一种综合的语义数据建模概念集, 以便对现实世界中复杂应用的实体和联系建模。因此, 面向对象数据库系统首先是一个数据库系统, 具备数据库系统的基本功能, 其次是一个面向对象的系统, 针对面向对象的程序设计语言的永久性对象存储管理而设计的, 充分支持完整的面向对象概念和机制。

(4) 多媒体数据库。多媒体数据库系统 (Multi-media DataBase System, MDBS) 是数据库技术与多媒体技术相结合的产物。在许多数据库应用领域中, 都涉及大量的多媒体数据。多媒体数据比传统的数字、字符等格式化数据复杂。多媒体数据库系统具有数据量大、结构复杂、音画时序性、数据传输的连续性等特点。从实际应用的角度考虑, 多媒体数据库管理系统 (MDBMS) 应具有以下基本功能:

1) 有效地表示多种媒体数据, 对不同媒体的数据如文本、图形、图像、声音等能够按应用的不同, 采用不同的表示方法。

2) 能够处理各种媒体数据, 正确识别和表现各种媒体数据的特征, 各种媒体间的空间或时间关联。

3) 能够像其他格式化数据一样对多媒体数据进行操作, 包括对多媒体数据的浏览、查询检索, 对不同的媒体提供不同的操纵, 如声音的合成、图像的缩放等。

4) 具有开放功能, 提供多媒体数据库的应用程序接口等。

(5) XML 数据库。随着 Internet 的迅速发展, Web 上各种半结构化、非结构化数据源已经成为重要的信息来源。XML (eXtended Markup Language) 意为可扩展的标记语言, 用户可以定义自己的标记, 用来描述文档的结构。XML 是 W3C (World Wide Web Consortium) 在 1998 年制定的一项标准, 用于网上数据交换。XML 是在标准通用标记语言 SGML 和超文本标记语言 HTML 的基础上发展起来的, 它汲取了两者的优点并克服了 SGML 过于简单和 HTML 局限性等缺点, 具有更多的结构和语义、可扩展性、自描述性、数据与显示分离和简洁性等特点。

(6) 数据仓库与联机分析处理。信息技术的高速发展, 数据和数据库在急剧增长, 数据库应用的规模、范围和深度不断扩大, 一般的事务处理已不能满足应用的需要, 企业界需要在大量信息数据基础上的决策支持 (Decision Support, DS) 以及数据仓库 (Data Warehousing, DW) 技术的兴起满足了这一需求。数据仓库作为决策支持系统 (Decision Support System, DSS) 的有效解决方案, 涉及 3 方面的技术内容: 数据仓库技术、联机分析处理 (On-Line Analysis Processing, OLAP) 技术和数据挖掘 (Data Mining, DM) 技术。

1.1.4 数据模型

1. 实体及其属性

所谓实体是指客观存在的并可相互区别的任何事物，如工厂、设备、产品、客户等。

所谓属性是指实体所代表的某特定事物所具有的某方面的特征。例如，把人作为一个实体，在学生档案中，用学号、姓名、性别、籍贯、出生日期等几个属性来刻画学生的特征。

实体与属性有型与值之分，所谓型是指结构或框架，值是指在结构约束下的具体取值。例如，表 1-1 所示的学生档案表。

表 1-1 学生档案表

学号	姓名	性别	籍贯	出生日期	免试否	入学分	简历	照片
010401	张清	男	合肥	85/08/21	免试		三好	
010402	李一文	女	芜湖	85/12/21		578		
010403	胡同木	男	滁州	86/03/18		546		

属性型

实体型

实体值

属性值

2. 数据模型

数据模型是指反映实体及其实体间联系的数据组织的结构和形式。在现实世界中，实体之间的联系可分为 3 种模型：“一对一”的联系、“一对多”的联系、“多对多”的联系，如图 1-2 所示。

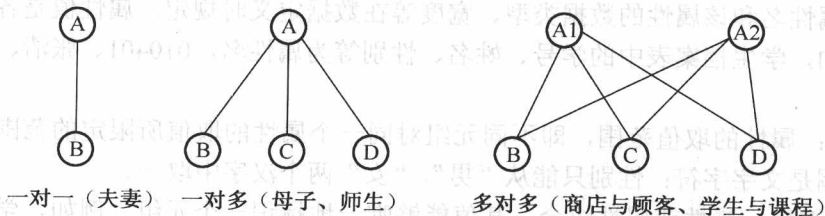


图 1-2 实体之间的联系

将实体之间的这种联系经过适当的转换，用一定的数据结构表示出来，就得到了数据模型，目前成熟地应用在数据库技术中的数据模型可分为以下 3 种基本模型：

(1) 层次模型。层次模型具有以下特点：①有且仅有一个根结点无双亲；②其他结点有且仅有一个双亲。它只能反映实体间一对多的联系。

(2) 网状模型。网状模型具有以下特点：①有一个以上结点无双亲；②至少有一个结点有多于一个双亲。它能反映实体间的“多对多”的联系，但结构与使用均比较复杂。

(3) 关系模型。关系模型可以看作用一张二维表格来描述实体间的联系。表格的栏目称为关系框架，也称为关系模式，相当于记录型，每一栏目称为属性，又称为字段名。表格中填满的全部数据称为关系。关系模型既能反映属性间一对一的联系，又能反映属性间一对多的联系，还能反映属性间多对多的联系。关系模型具有以下特点：①数据结构简单，概念

清楚,符合习惯;②能直接反映实体间的三种联系;③格式单一,一律为表格框架,通过公共属性可建立关系之间的联系;④具有严格的理论基础。

1.2 关系数据库

任务的提出

既然关系模型可以看作一张二维表格,它与数据库有什么联系以及遵循哪些运算,自然是接着要讨论的问题。本任务知识点主要包括以下内容:

- (1) 关系数据库。
- (2) 关系运算。
- (3) 关系的完整性约束。

1.2.1 关系数据库

1. 关系术语

(1) 关系:一个关系就是一张二维表,每个关系有一个关系名。在计算机里,一个关系可以存储为一个文件,在 Visual FoxPro 中称为表文件。

(2) 元组:二维表中水平方向的行称为元组,每一行是一个元组。元组对应存储文件中的一个具体记录。例如,表 1-1 所示的学生档案表对应的关系包括 3 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,与前面讲的实体属性相同。属性名和该属性的数据类型、宽度等在数据定义时规定。属性值是各个记录中的字段值。例如,学生档案表中的学号、姓名、性别等为属性名,010401、张清、男、合肥等是属性值。

(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。例如,姓名的取值范围是文字字符;性别只能从“男”、“女”两个汉字中取一。

(5) 关键字:属性或属性组合,其值能够唯一地标识一个元组。例如,学生档案表中的学号可以作为标识关键字;如果绝对没有重名学生,则姓名也能唯一地标识一个元组。由于具有某一性别的可能不止一个人,性别就不能作为标识关键字。

(6) 关系模式:对关系的描述称为关系模式,格式为:

关系名(属性名 1,属性名 2,...,属性名 n)

一个关系模式对应一个关系的结构。

(7) 关系数据库:关系数据库是以二维表格的形式组织或存放数据的数据库,它是若干关系的集合。Visual FoxPro 就是关系数据库管理系统的一种。在 Visual FoxPro 中将关系叫做数据库表或表文件,关系名称为表名,元组称为记录,属性名称为字段名。

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关系模式的集合。

2. 关系的性质

既然关系表现为二维表,则可通过二维表研究关系的性质:

- (1) 关系不允许“表中套表”，即表中元组分量必须是原子的。
- (2) 在同一个关系中不能出现相同的属性名。
- (3) 关系中不允许有完全相同的元组。
- (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。
- (5) 在一个关系中列的次序无关紧要。

1.2.2 关系运算

1. 传统的集合运算

传统的关系运算主要指并、交、差，当用于关系时，要求参加运算的两个关系是相容的，即两关系度数相同，相应属性取自同一个域。

设 R 和 S 是相容二关系，则

- (1) 并：由属于 R 或属于 S 或同时属于 R 和 S 的元组构成的集合，记为 $R \cup S$ 。
- (2) 差：由属于 R 而不属于 S 的所有元组组成的集合，记为 $R - S$ 。
- (3) 交：由同时属于 R 和 S 的元组组成的集合，记为 $R \cap S$ 。

【例 1.1】设 R 和 S 如表 1-2 所示，则 $R \cup S$ 、 $R - S$ 、 $R \cap S$ 结果仍为关系。

表 1-2 3 种集合基本运算

R			S			$R \cup S$			$R - S$			$R \cap S$		
X	Y	Z	X	Y	Z	X	Y	Z	X	Y	Z	X	Y	Z
a	3	e	b	1	d	a	3	e	a	5	a	a	3	e
b	1	d	c	2	c	b	1	d				b	1	d
a	5	a	a	3	e	a	5	a						
						c	2	c						

2. 专门的关系运算

专门的关系运算包括选择、投影和连接。

(1) 选择。单目运算，它是从给定的关系 R 中选择满足条件 F 的那些元组，组成一个新的关系 S，记作 $S = \sigma_F(R)$ 。

【例 1.2】从表 1-1 所示的学生档案关系中取出年龄不超过 19 岁的学生，构成一新的关系 S，操作为： $S = \sigma_{\text{性别}="男"}(\text{学生档案})$ ，操作结果如表 1-3 所示。从表 1-3 可以看出，该关系 S 和原关系 R 的属性名完全相同。

表 1-3 学生档案关系的选择 S

学号	姓名	性别	籍贯	出生日期	免试否	入学分数	简历	照片
010401	张清	男	合肥	85/08/21	免试		三好	
010403	胡同木	男	滁州	86/03/18		546		

(2) 投影。单目运算，它是从给定的关系 R 中将某些列划掉，在剩余的若干列，即投影属性 A 所构成的关系中去掉重复元组后所构成新关系 S，记作 $S = \Pi_A(R)$ 。

【例 1.3】对学生档案关系的姓名、性别、出生日期 3 个属性进行投影,得到关系 S,其操作为: $S = \Pi_{\text{姓名、性别、出生日期}}(\text{学生档案})$, 投影得到的新关系 S 如表 1-4 所示。

表 1-4 学生档案关系的投影 S

姓名	性别	出生日期
张清	男	85/08/21
李一文	女	85/12/21
胡同木	男	86/03/18

(3) 连接。双目运算,它是从两个关系的充分组合中选取满足条件 F 的元组形成新的关系 T,可以记作 $T = R \underset{F}{\bowtie} S$ 。

【例 1.4】已知表 1-5 和表 1-6 所示的关系 R、S,则满足条件“奖金>补贴”的连接运算为 $T = R \underset{\text{奖金}>\text{补贴}}{\bowtie} S$,得到的新关系 T 如表 1-7 所示。

表 1-5 关系 R

姓名	工资	奖金
王君	300	500
李江明	225	100
林莉	500	90
汪峰	450	96
舒新	580	150

表 1-6 关系 S

工资	补贴
225	400
239	100
500	500
750	900
630	800

表 1-7 关系 T

姓名	工资	奖金	工资	补贴
王君	300	500	225	400
王君	300	100	239	100
舒新	580	150	239	100

关系 T 是将关系 R 中的王君的奖金分别与关系 S 中的补贴比较,满足条件的只有第一、第二两行,记录到关系 T 中;再将李江明的奖金与关系 S 中的补贴比较,没有满足条件的行,T 中不记录……。

如果条件 F 是一个相等的条件,就是等值连接的情况,称之为自然连接。

【例 1.5】对上例中的关系 R 和 S,自然连接运算 $T = R \bowtie S$ 的操作结果如表 1-8 所示,它是表 1-5 与表 1-6 中两工资相等的操作结果。

表 1-8 关系 T

姓名	工资	奖金	补贴
李江明	225	100	400
林莉	500	90	500

1.2.3 关系的完整性约束

关系模型由三部分组成：数据结构（即前面所讲的关系模式）、关系运算和关系的完整性。完整性是指数据的正确性和相容性，是为防止数据库中存在不符合语义的数据，防止错误信息输入和输出。关系的完整性有实体完整性、域完整性和参照完整性三种。前两种是关系模型必须满足的完整性约束。

1. 实体完整性

实体完整性约束是指关系中元组的主关键字的值不能为空值（空值是“不知道”或“无意义”的值）。如果出现空值，那么主关键字值就起不到唯一标识元组的作用，关系数据库通过主关键字查找元组的操作将无法正确完成。

2. 域完整性

域完整性约束是指不同元组对同一个属性的取值所限定的范围。它通过指定不同的宽度说明不同范围的数据类型，从而可以限定字段的取值类型和取值范围。但这些对域完整性还远远不够，还可以用一些域约束规则来进一步保证域完整性。域约束规则也称作字段有效性规则，主要用于数据输入正确性的检验。

3. 参照完整性

参照完整性（或叫引用完整性）规定不允许引用不存在的元组。确切地说，参照完整性指的是一个关系的外关键字的取值必须是与之关联的关系中主关键字的有效值或空值。

1.3 Visual FoxPro 系统概述



学习了前面的内容以后，大家关心的问题可能是如何建立表文件，对它们如何进行操作。当然这些还需对 Visual FoxPro 的使用环境有一个基本认识以后才能实现。

1.3.1 Visual FoxPro 的发展历史与系统特点

数据库理论的研究起源于 20 世纪 60 年代，在 70 年代后期进入较为成熟的阶段，随着 80 年代初 IBM/PC 及其兼容机的广泛使用，数据库产品的代表作之一，Ashton-Tate 公司开发的 dBASE 很快进入微机世界，成为受到普遍欢迎的数据库管理系统。由于它易于使用、功能较强，很快处于 80 年代中期微机数据库管理系统的主导地位（极盛时期曾在个人计算机的数据库管理系统市场上的占有率高达 80%~85%）。继 dBASE II 之后，dBASE III、dBASE III PLUS 及 dBASE IV（通称为 XBase）相继诞生，其功能逐渐增强。

但是，dBASE 存在的一些缺陷使其应用受到越来越大的限制。首先，它运行速度慢，这在建立大型数据库时显得尤为突出。其次，早期的 dBASE 不带编译器，是解释执行的，后来虽然增加了编译器，但编译与解释执行时存在许多差异。再就是它的设计标准，随着 dBASE 增强版本的出现，由于各版本之间不兼容，其标准变得越来越模糊，Ashton-Tate 公