

Visual FoxPro数据库 程序设计教程

主 编	章 伟	杨正武
副主编	梁 敏	伍永锋
	周 海	廖文婧



科学出版社

014058934

TP311.138FOXP
341

内 容 简 介

本书结合 Visual FoxPro 数据库系统的特点，以 Visual FoxPro 6.0 为平台，深入浅出地介绍了数据库设计、数据库系统、数据库应用、数据库安全等方面的知识。全书共分 10 章，第 1 章介绍数据库系统概论，第 2 章介绍 Visual FoxPro 数据库系统，第 3 章介绍数据库设计，第 4 章介绍数据库系统，第 5 章介绍数据库应用，第 6 章介绍数据库安全，第 7 章介绍数据库系统，第 8 章介绍数据库应用，第 9 章介绍数据库安全，第 10 章介绍数据库系统。本书可作为高等院校计算机专业及相关专业的教材，也可供从事数据库工作的工程技术人员参考。

Visual FoxPro 数据库 程序设计教程

主 编 章 伟 杨正武
副主编 梁 敏 伍永锋
周 海 廖文婧



科学出版社

北京



北航

C1746226

TP311.138FoxP

341

1014028334

内 容 简 介

本书结合作者多年数据库应用课程的教学经验,并且兼顾“全国计算机等级考试二级 Visual FoxPro 数据库程序设计考试大纲(2013 版)”的要求编写而成。

本书以 Visual FoxPro 6.0 为基础,介绍关系数据库管理系统的基础理论及系统开发技术。主要内容包括关系数据库管理系统的基本概念、Visual FoxPro 基础知识、数据库基础操作、结构化查询语言 SQL、查询与视图、报表设计与应用、结构化程序设计、表单设计与应用、菜单设计与应用等。各章实例均参照全国计算机等级考试相关题目设计,另有《Visual FoxPro 数据库程序设计实践与题解》(主编梁敏)辅助配套教材。

本书可作为普通高等院校各专业计算机公共课教材,也可作为全国计算机等级考试(二级 Visual FoxPro)的培训教材,还可作为数据库应用系统开发人员的参考书。

图书在版编目(CIP)数据

Visual FoxPro 数据库程序设计教程/章伟,杨正武主编.—北京:科学出版社,2014.8

ISBN 978-7-03-041443-4

I. ①V… II. ①章… ②杨… III. ①关系数据库系统—程序设计—教材
IV. ①TP311.138

中国版本图书馆 CIP 数据核字(2014)第 167262 号

责任编辑:李 清 张丽花/责任校对:郭瑞芝

责任印制:闫 磊/封面设计:迷底书装

科学出版社 出版

北京东黄城根北街 16 号

邮政编码:100717

<http://www.sciencep.com>

北京市文林印务有限公司 印刷

科学出版社发行 各地新华书店经销

*

2014 年 8 月第 一 版 开本:787×1092 1/16

2014 年 8 月第一次印刷 印张:13

字数:341 000

定价:30.00 元

(如有印装质量问题,我社负责调换)

前 言

Visual FoxPro 是 Microsoft 公司推出的面向对象的数据库管理系统。它具有丰富的开发工具、较高的处理速度、友好的交互界面、完备的兼容性及面向对象等特点。Visual FoxPro 采用可视化的面向对象的程序设计方法,大大简化应用程序的开发过程,从而得到了广泛的应用,成为中小型数据库应用系统的典型程序开发工具,适合中小型企业用于管理信息系统的开发。

本书以 Visual FoxPro 6.0 为基础,覆盖全国计算机等级考试二级考试大纲(Visual FoxPro 程序设计),并结合高等学校本科教学的实际要求,力求全面细致地讲述 Visual FoxPro 的基础知识和应用程序设计方法。本书除了在内容上全面覆盖全国计算机等级考试二级考试(Visual FoxPro 程序设计)外,考虑到学生参加全国计算机等级考试的实际需要,在例题和习题的选择上尽可能地贴近全国计算机等级考试二级考试真题,以期对学生有所帮助。

本书共分 9 章内容:概述、Visual FoxPro 基础知识、数据库基础操作、结构化查询语言 SQL、查询与视图、报表设计与应用、结构化程序设计、表单设计与应用、菜单设计与应用。基本涵盖高等院校“Visual FoxPro 数据库及应用”课程的教学大纲和《全国计算机等级考试二级考试大纲(Visual FoxPro 程序设计)》的要求。本书提供的大量例题和实验是“积木式”结构,作者强调实践环节和培养学生的应用能力,例题和习题均是在计算机二级考试中总结、精选得到的有代表性的题目。

本书编写人员分工为:章伟(第 4、7 章)、梁敏(第 1、2 章)、伍永锋(第 8 章)、周海(第 3 章)、廖文婧(第 5、6、9 章),全书由章伟老师统稿。

在本书的编写过程中,始终得到贵州财经大学教务处领导、信息学院领导及计算机基础教研室全体老师大力支持,在此表示衷心感谢。

由于时间仓促,水平有限,书中可能存在疏漏之处,恳请广大读者提出宝贵意见。

编 者

2014 年 6 月

目 录

前言

第 1 章 概述1

1.1 数据库系统的基本概念1

1.1.1 数据、数据库、数据库管理

系统和数据库系统1

1.1.2 数据管理技术的发展2

1.1.3 数据库的特点3

1.1.4 数据库系统体系结构3

1.2 数据模型4

1.3 关系数据理论基础7

1.3.1 基本的关系术语7

1.3.2 关系模型的完整性约束条件8

1.3.3 关系数据操纵9

第 2 章 Visual FoxPro 基础知识10

2.1 Visual FoxPro 简介10

2.1.1 Visual FoxPro 集成环境的

使用10

2.1.2 文件类型与创建12

2.1.3 Visual FoxPro 系统环境的

配置13

2.2 Visual FoxPro 的数据类型15

2.3 常量与变量18

2.3.1 常量18

2.3.2 变量20

2.4 运算符与表达式25

2.4.1 数值运算符与数值表达式25

2.4.2 字符运算符与字符表达式26

2.4.3 日期、日期时间运算符与

日期、日期时间表达式26

2.4.4 关系运算符与关系表达式27

2.4.5 逻辑运算符与逻辑表达式27

2.5 函数28

2.5.1 数值函数29

2.5.2 字符函数32

2.5.3 日期和时间函数35

2.5.4 数据类型转换函数37

2.5.5 数据库与表函数39

2.5.6 其他函数41

第 3 章 数据库基础操作46

3.1 Visual FoxPro 项目管理器46

3.1.1 项目文件的建立46

3.1.2 项目管理器的界面46

3.1.3 项目管理器的使用48

3.2 Visual FoxPro 数据库49

3.2.1 基本概念49

3.2.2 建立数据库50

3.2.3 打开数据库50

3.2.4 修改数据库51

3.2.5 关闭数据库51

3.2.6 删除数据库51

3.2.7 数据库实例操作51

3.3 表结构53

3.3.1 数据库表结构的建立53

3.3.2 数据库表结构的修改54

3.3.3 自由表56

3.3.4 数据库表操作实例57

3.4 表记录61

3.4.1 在浏览器中操作表61

3.4.2 操作表的相关命令63

3.4.3 表记录定位63

3.4.4 表记录实例操作64

3.5 索引65

3.5.1 基本概念65

3.5.2 索引的建立66

3.5.3 索引的使用	67	5.1.2 查询设计实例	118
3.5.4 索引的删除	68	5.2 视图	124
3.5.5 索引的相关命令	68	5.2.1 视图的概念	124
3.6 数据完整性	69	5.2.2 使用 SQL 命令建立视图	124
3.6.1 实体完整性	69	5.2.3 使用视图设计器建立视图	126
3.6.2 数据库表高级属性及域完整性设置	69	第 6 章 报表设计与应用	128
3.6.3 永久联系与参照完整性	72	6.1 报表概述	128
3.7 工作区	75	6.1.1 报表的常规布局	128
3.7.1 基本概念	75	6.1.2 报表向导	128
3.7.2 工作区操作命令	75	6.2 报表设计器	133
3.7.3 数据工作期	76	6.2.1 报表带区	134
3.7.4 表之间的关联	77	6.2.2 工具栏的使用	134
3.8 附表	78	6.2.3 预览报表	135
第 4 章 结构化查询语言 SQL	81	6.3 报表设计实例	135
4.1 SQL 概述	81	第 7 章 结构化程序设计	143
4.2 数据定义语言	81	7.1 程序文件	143
4.2.1 建立表的命令(CREATE)	81	7.1.1 程序的书写规则	143
4.2.2 修改表结构的命令(ALTER)	83	7.1.2 程序文件的建立、编辑和运行	143
4.2.3 删除表的命令(DROP)	85	7.1.3 程序设计的原则和方法	144
4.3 数据操纵语言	85	7.2 程序的基本结构	144
4.3.1 追加记录的命令(INSERT)	85	7.2.1 顺序结构	144
4.3.2 删除记录的命令(DELETE)	86	7.2.2 分支(选择)结构	145
4.3.3 更新记录的命令(UPDATE)	86	7.2.3 循环结构	148
4.4 数据查询	87	7.3 子程序、过程与自定义函数	151
4.4.1 投影查询	88	7.3.1 子程序	152
4.4.2 条件查询	90	7.3.2 过程	152
4.4.3 统计和分组统计	93	7.3.3 自定义函数	153
4.4.4 查询排序	100	7.3.4 内存变量的作用域和参数传递	154
4.4.5 查询结果的处理	102	第 8 章 表单设计与应用	156
4.4.6 多表查询	104	8.1 面向对象基本概念	156
4.4.7 嵌套查询	109	8.1.1 对象与类	156
4.4.8 超连接查询	113	8.1.2 容器与控件	158
4.4.9 集合的并运算查询	116	8.1.3 对象的引用	158
第 5 章 查询与视图	118	8.1.4 事件与方法	159
5.1 查询	118		
5.1.1 查询设计器	118		

8.2 表单的创建与运行	160	8.4.7 列表框(List)	173
8.2.1 创建表单	160	8.4.8 组合框(Combo)	175
8.2.2 运行表单	161	8.4.9 表格(Grid)	176
8.3 表单的操作	161	8.4.10 页框(PageFrame)	181
8.3.1 表单设计器	161	8.4.11 计时器(Timer)	183
8.3.2 常用表单属性	162	第9章 菜单设计与应用	185
8.3.3 常用表单事件与方法	163	9.1 菜单概述	185
8.3.4 数据环境	163	9.1.1 菜单的结构	185
8.4 常用控件	165	9.1.2 菜单的类型	185
8.4.1 标签	165	9.2 下拉式菜单设计	186
8.4.2 文本框(Text) 和编辑框 (EditBox)	166	9.2.1 菜单设计的一般步骤	186
8.4.3 命令按钮(Command) 和命令 按钮组(CommandGroup)	167	9.2.2 “菜单设计器” 窗口	187
8.4.4 选项按钮组(OptionGroup)	169	9.2.3 “显示” 菜单	191
8.4.5 复选框(Check)	170	9.3 快捷菜单设计	193
8.4.6 微调按钮(Spinner)	172	9.4 为顶层表单添加菜单	197
		参考文献	200

第1章 概述

信息资源已经成为各个部门的重要财富和资源。数据库能够有效、合理地存储各种数据,为有关应用准确、快速地提供有用的信息,是数据处理的重要工具,是管理信息系统(MIS)、办公信息系统(OIS)和决策支持系统(DSS)等应用系统的核心部分。数据库技术是计算机领域中最重要技术之一,其应用已渗透到人类社会各个领域,并正在改变着人们的生活方式和工作方式。因此,我们有必要学习和掌握数据库系统的原理和技术,用以解决各种计算机应用中的实际问题。

1.1 数据库系统的基本概念

在系统介绍数据库的基本概念之前,首先介绍一些数据库最常用的术语和基本概念。

1.1.1 数据、数据库、数据库管理系统和数据库系统

1. 数据

信息与数据是两个既有区别又有联系的概念。归纳起来,信息、数据及其联系可以这样来定义。

1) 信息

信息(Information)是指现实世界事物的存在方式或运动状态的反映。通俗地讲,信息是经过加工并对人类社会实践和生产以及经营活动产生决策影响的数据。

2) 数据

描述事物的符号记录称为数据(Data)。描述事物的符号可以是数字,也可以是文字、图形、图像、声音、语言等,数据有多种表现形式,它们都可以经过数字化后存入计算机。

3) 信息与数据的关系

不是所有数据都能成为信息,只有经过提炼和浓缩之后,具有新知识的数据才能成为信息。不经过加工处理的数据只是一堆死材料,对人类活动产生不了决策作用。数据经过加工处理之后所得到的信息,仍然以数据的形式出现,此时的数据是信息的载体;而信息是数据的内涵,是数据的语义解释。

2. 数据库

数据库(DataBase, DB)是长期存储在计算机内的、有组织的、可共享的数据集合。数据库中的数据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,并可为各种用户共享。

3. 数据库管理系统

数据库管理系统 (DataBase Management System, DBMS) 是数据库系统中专门用于数据管理的软件, 是用户与数据库的接口。它的主要功能包括以下几个方面。

1) 数据库定义功能

数据库管理系统提供数据描述语言 (Data Defined Language, DDL) 及其翻译程序, 用于定义数据库结构 (模式及模式间映射)、数据完整性和保密性约束等。

2) 数据库操纵功能

数据库管理系统提供数据操纵语言 (Data Manipulation Language, DML) 及其翻译程序, 用于实现对数据库数据的查询、插入、更新和删除等操作。

3) 数据库运行和控制功能

包括数据安全性控制、数据完整性控制、多用户环境的并发控制等。

4) 数据库维护功能

包括数据库数据的载入、转储和恢复, 数据库的维护和数据库的功能及性能分析和监测等。

4. 数据库系统

数据库系统 (DataBase System, DBS) 是指在计算机系统中引入数据库后的系统, 一般由数据库、数据库管理系统 (及其开发工具)、应用系统、数据库管理员和用户构成。应当指出的是, 数据库的建立、使用和维护等工作只靠一个 DBMS 远远不够, 还要有专门的人员来完成, 这些人员被称为数据库管理员 (DataBase Administrator, DBA)。

1.1.2 数据管理技术的发展

数据管理技术是指对数据进行分类、组织、编码、存储、检索和维护的技术。在计算机环境下, 数据管理技术经历了从低级到高级的三个发展阶段。

1. 人工管理阶段 (20 世纪 50 年代中期以前)

在人工管理阶段, 数据处理的主要任务是数据量很少的科学计算, 没有专门的软件对数据进行管理, 在程序设计中, 既要考虑程序的处理过程, 又要考虑数据的定义和组织, 程序和数据总是联系在一起的。

人工管理阶段的特点: 数据不保存, 程序运行结束后数据就丢失; 数据由应用程序自己管理, 没有相应的软件系统负责数据的管理工作; 数据不能共享, 程序和数据是一个整体, 一个程序中的数据无法被其他程序共享; 数据不具有独立性, 一旦修改了数据的存储结构, 则其程序也必须修改。

2. 文件系统管理阶段 (20 世纪 50 年代后期至 60 年代中期)

在文件系统阶段, 数据处理的主要任务是科学计算和简单的数据管理, 有了专门管理数据的软件, 即文件系统。数据已经从程序中分离出来, 组成相互独立的数据文件并能保存, 程序和数据之间具备一定的独立性。

文件系统的特点: 数据能长期保存, 能反复地进行查询、修改、插入、删除操作;

由文件系统进行数据管理, 程序和程序之间由软件提供的存取方法, 如函数调用等方式进行转换, 使数据和程序之间有了一定的独立性。文件系统存在的不足: 数据的冗余度大, 不同的应用程序使用相同的数据时, 就要建立不同的数据文件, 而内容却是相同的; 数据的独立性差, 数据的组织形式依赖于应用程序, 相互有影响, 不利于系统移植、系统扩充、系统维护等工作; 缺乏对数据的统一控制管理, 数据的完整性和数据的安全性很难得到保证, 各个数据文件需要有管理软件实施统一控制管理; 不支持对文件的并发访问。

3. 数据库系统管理阶段(20世纪60年代后期开始)

在数据库系统阶段, 数据处理的主要任务是各种科学计算和大量的、复杂的数据管理, 有了统一管理数据的专门软件系统, 即数据库管理系统。数据与程序已经完全独立, 大大降低了数据的冗余度, 实现了多用户、多应用的数据共享。

1.1.3 数据库的特点

数据库系统的出现, 是计算机数据处理技术的重大进步, 它具有以下基本特点。

1. 数据结构化

在数据库系统中, 数据是按照特定的模型进行组织的, 数据文件中记录的内容, 不仅能描述数据本身, 而且能表示数据之间的联系。数据库系统实现整体数据的结构化, 这种特征能够反映现实世界的客观联系, 能适应大批量数据管理的客观需要。

2. 数据共享、冗余度低

数据共享是数据库系统的目的, 也是它的重要特点。在数据库系统中, 数据是面向整个系统的, 可为所有访问系统的用户共享。数据冗余是指各数据文件中有相互重复的数据。从理论上讲, 可以消除冗余, 但实际上, 常常允许部分冗余存在, 以提高检索速度。

3. 数据独立性高

在数据库系统中, 数据库的建立独立于程序, 数据库系统通过三级模式和两种映像功能, 使数据具有物理独立性和逻辑独立性。物理独立性是指当数据的存储结构(也称存储模式或内模式)改变时, 通过映像, 数据的逻辑结构(也称逻辑模式或模式)不变, 从而不必修改应用程序。逻辑独立性是指当数据的逻辑结构改变时, 通过映像, 数据的用户模式(也称子模式或外模式)不变, 从而也不必修改应用程序。

4. 数据库管理系统(DBMS)统一管理和控制

通过 DBMS 软件包统一管理数据, 实现多用户的数据共享和并发操作, 并确保数据的安全性和数据的完整性, 包括数据库恢复的功能。

1.1.4 数据库系统体系结构

数据库的体系结构是数据库系统的一个总框架。尽管实际数据库软件产品种类繁多, 使用的数据库语言各异, 基础操作系统不同, 采用的数据结构模型相差甚大, 但是绝大多

数据库系统在总体结构上都具有三级模式的结构特征。数据库的三级模式结构由外模式、模式和内模式组成,如图 1-1 所示。

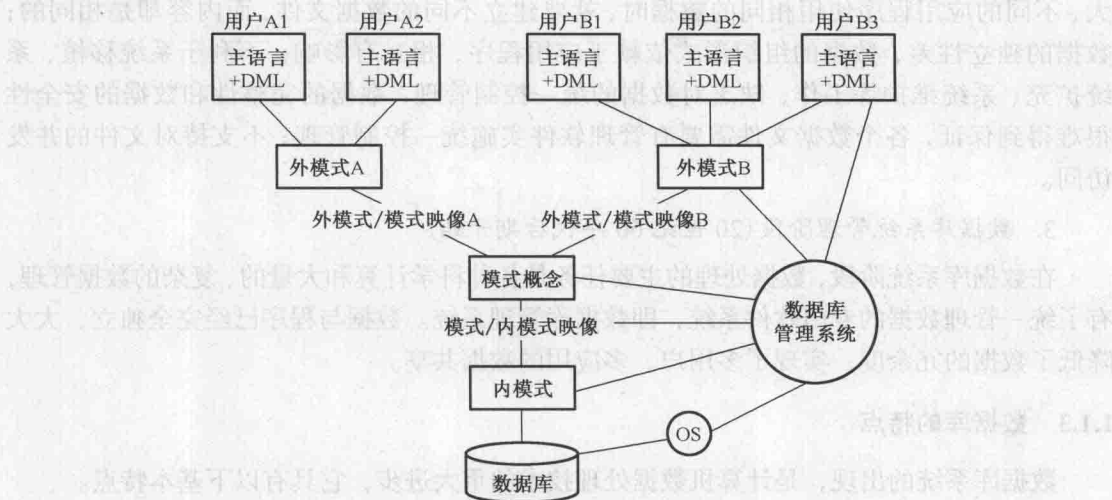


图 1-1 数据库系统的体系结构

(1) 外模式: 又称子模式或用户模式, 是模式的子集, 是数据的局部逻辑结构, 也是数据库用户看到的数据视图。

(2) 模式: 又称逻辑模式或概念模式, 是数据库中全体数据的全局逻辑结构和特性的描述, 也是所有用户的公共数据视图。

(3) 内模式: 又称存储模式, 是数据在数据库系统中的内部表示, 即数据的物理结构和存储方式的描述。

数据库系统的三级模式是对数据的三级抽象。为了实现三个抽象层次的转换, 数据库系统在三级模式中提供了两次映像: 外模式/模式映像和模式/内模式映像。所谓映像, 就是存在某种对应关系。

外模式到模式的映像, 定义了外模式与模式之间的对应关系。模式到内模式的映像, 定义了数据的逻辑结构和物理结构之间的对应关系。正是由于这两级映像, 使数据库管理的数据具有两个层次的独立性: 物理独立性和逻辑独立性。

1.2 数据模型

模型是对现实世界某个事物特征的模拟和抽象。由于用计算机来研究处理现实世界的具体事物时, 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。所以, 数据模型是抽象、表示、处理现实世界中事物的基本工具。

1. 信息的三个世界

信息的三个世界是指现实世界、信息世界和计算机世界。现实世界是指客观存在的事物, 它是信息之源, 是设计和建立数据库的出发点, 也是使用数据库的最终归宿。信息

世界, 又称观念世界, 是现实世界中的客观事物在人头脑中的反映, 客观事物在信息世界中称为实体, 而反映事务间关系的称为实体模型或概念模型。计算机世界是指信息世界中的信息数据转化成能被计算机处理的数据, 又称数据世界。实体模型在数据世界中以数据模型描述。现实世界、信息世界和计算机世界这 3 个领域是由客观到认识、由认识到使用管理的 3 个不同层次, 后一领域是前一领域的抽象描述。现实世界的事物及联系, 通过抽象成为信息世界的概念模型, 而概念模型经过数据化处理转换为数据模型, 其变化关系如图 1-2 所示。

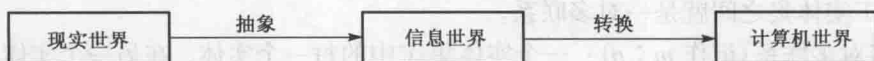


图 1-2 3 个世界的变化关系

2. 概念模型

概念模型是数据库设计人员在认识现实世界中的实体及实体间联系后进行的一种抽象, 是用户与数据库设计人员之间进行交流的语言。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, 数据模型是借助概念模型(或信息模型)转化而来的。概念模型中涉及以下概念。

(1) 实体: 客观存在并且可以相互区分的事物称为实体。它可以指人, 如一名教师、一个学生等; 也可以指物, 如一本书、一张桌子、一块黑板等; 也可以指抽象的事件, 如借书、奖励、交通法规等。它还可以指事物与事物之间的联系, 如学生选课、客户订货、顾客购物等。

(2) 属性与属性值: 用于描述实体的特性称为实体的属性, 如职工实体用职工号、姓名、性别、出生日期等若干个属性描述。属性的具体取值称为属性值, 用于刻画一个实体, 如属性值的组合(10023, 张红芳, 女, 1965-10-24)就描述了一个具体的职工。每个属性的特定取值范围称为值域, 如性别的值域为(男, 女)。

(3) 实体型与实体值: 由上可见, 属性值所组成的集合表征一个实体, 相应的这些属性的集合表征了一种实体的类型, 称为实体型, 如(职工号, 姓名, 性别, 出生日期)表征职工实体的实体型。实体值是指某个具体实体的取值, 如(10023, 张红芳, 女, 1965-10-24)就是一个实体值。

(4) 实体集: 同种类型实体的集合称为实体集。例如, 全体记者就组成一个实体集。在 Visual FoxPro 中, 用“表”表示同一类实体, 即实体集, 用“记录”表示一个具体的实体, 用“字段”表示实体的属性, 表的结构对应于实体型。

(5) 码: 在众多属性中能够唯一标识或确定一个实体的属性或属性组称为实体的码, 如学生实体的码应当是“学号”。

(6) 联系: 联系是指反映现实世界事物之间的相互关系。一个实体内部各属性之间的相互联系称为实体的内部联系。各实体集之间的相互联系称为实体的外部联系。这些联系可以分为一对一、一对多、多对多 3 种类型。

① 一对一联系(记作 1:1): 一个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；反之亦然，称这两个实体集之间存在着一对一联系。比如，一个企业只有一位总经理，并且一位总经理只能管理一个企业，所以企业实体集和总经理实体集之间就是一对一联系。

② 一对多联系(记作 $1:n$)：一个实体集 X 中的每一个实体，在另一个实体集 Y 中能够找到多个可以与它相对应的实体；反之，在另一个实体集 Y 中的每一个实体，却只能在实体集 X 中找到一个可以与它相对应的实体，称这两个实体集之间存在着一对多联系。比如，一个企业有许多职工，但每一个职工只能工作(含人事关系)在一个企业，所以企业实体集和职工实体集之间就是一对多联系。

③ 多对多联系(记作 $m:n$)：一个实体集 X 中的每一个实体，在另一个实体集 Y 中能够找到多个可以与它相对应的实体；反之，在另一个实体集 Y 中的每一个实体，也能在实体集 X 中找到多个可以与它相对应的实体，称这两个实体集之间存在多对多联系。比如，一个学生可以选修多门课程，而每一门课程可以被多个学生选修，所以学生实体集和课程实体集之间就是多对多联系。

描述概念模型的方法很多，其中最常用的是实体-联系方法(Entity-Relationship Approach)，简称 E-R 方法。E-R 方法的规则是：用长方形表示实体，并在框内写上实体名；用椭圆表示实体属性，并用无向边(即直线)把实体与其属性连接起来；用菱形表示实体间的联系，菱形框内写上联系名；用无向边把菱形分别与相关的实体相连接，在无向边旁标上联系的类型($1:1$ 、 $1:n$ 、 $m:n$)。若实体之间的联系也具有属性，则把属性和菱形也用无向边连接上。上述 3 种联系的简单 E-R 方法表示如图 1-3 所示。

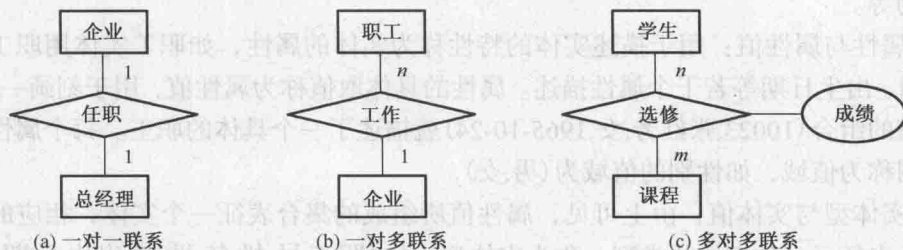


图 1-3 简单 E-R 方法表示

3. 逻辑数据模型

逻辑数据模型是数据库系统的核心和基础，它描述了数据库中数据的整体结构。逻辑数据模型通常由数据结构、数据操作、数据完整性约束 3 部分组成，其中数据结构是对系统静态特性的描述，是逻辑数据模型中最重要的部分，因此人们一般以逻辑数据结构的类型来命名该数据模型。由于采用的数据模型不同，相应的数据库管理系统也就完全不同。在数据库系统中，常用的数据模型包括层次模型、网状模型和关系模型。

(1) 层次模型：层次模型是用倒置的树形结构来表示实体及其之间的联系。在这种模型中，数据结构从树根开始向树枝、树叶逐层展开，树中的每一个结点代表一个实体，连线则表示它们之间的关系。层次模型的特点是：有一个结点没有父结点，这个结点称为根结点；其他结点有且仅有一个父结点。

(2)网状模型：网状模型是用实体为结点的图来表示各实体及其之间的联系，它可以表示数据间的纵向关系和横向关系，呈现出一种交叉联系的网络结构。网状模型的特点是：可以有一个以上的结点无父结点；至少有一个结点有多个父结点。

(3)关系模型：关系模型是用二维表格来表示实体及其相互之间的联系。在关系模型中，把实体集看成一张二维表，通过相同关键字段实现表格间的数据联系。每一张二维表称为一个关系，每个关系均有一个名称即关系名。关系模型是目前比较流行的一种数据模型，如学生的课程学分情况如表 1-1 所示。

表 1-1 课程学分情况

课程号	课程名	学时数	学分
0001	大学英语	80	4
0002	日语	60	3
0003	计算机应用基础	64	3
0004	C 语言程序设计	80	4
0005	数据库系统	80	4
0006	高等数学	80	4
0007	概率论	40	2
0008	会计学	80	4
0009	西方经济学	80	4
0010	人力资源管理基础	80	4

4. 物理数据模型

物理数据模型用来描述数据的物理存储结构和存储方法，它与计算机存储器和操作系统密切相关，一般用户在数据库设计时不需要过多地考虑物理结构，数据库管理系统会自动进行处理。

1.3 关系数据理论基础

在数据库中，如果数据结构按照层次模型定义，则该数据库为层次数据库；如果数据结构按照网状模型定义，则该数据库为网状数据库；如果数据结构按照关系模型定义，则该数据库为关系数据库。Visual FoxPro 数据库管理系统是一种关系数据库管理系统，它能让用户以最有效的方式管理和处理大量的数据，实现数据的增加、修改、查询、删除操作，制作报表和标签，快速自行开发一个简单的应用系统。

1.3.1 基本的关系术语

1. 关系模型的数据结构

关系数据库系统是由许多不同的关系构成，其中的每一个关系就是一个实体，可以用一张二维表来表示。

2. 关系模型的基本术语

(1)关系：关系是指一张二维表，由行和列组成。每个关系都有一个关系名，并以表文件的形式保存，其扩展名为.dbf。

(2) 属性：属性是指一张二维表中的每一列，属性有属性名和属性值。在 Visual FoxPro 中，一个属性对应表文件中的一个字段，属性名对应字段名，属性值对应表文件中各个记录的字段值。

(3) 元组：元组是指一张二维表中的每一行的属性值。在 Visual FoxPro 中，一个元组对应表文件中的一条记录(记录值)。

(4) 框架：框架是指由属性名组成的表头。在 Visual FoxPro 中，框架对应表文件中的表的结构。

(5) 域：域是指每个属性的取值范围。

(6) 关键字：关键字是指具有唯一确定一个元组的属性或属性组。比如，学生的“学号”属性。一个关系中可以有多个关键字。

(7) 主关键字：主关键字是指当前被指定的那个关键字，并且其属性值不能取“空值”。

(8) 外部关键字：外部关键字是指一张二维表中的某个属性或属性组虽然不是所在表的关键字，但却是另一张二维表的关键字。

(9) 关系模式：关系模式是指对关系的描述，可表示成：关系名(属性1,属性2,属性3,...,属性n)。比如，课程(课程号,课程名,学时数,学分)。

3. 关系的基本特点

在关系模型中，每个关系都必须满足一定的要求，其基本特点是：关系必须规范化，属性不可再分割；在同一关系中不允许出现相同的属性名；在同一关系中不允许有完全相同的元组；在同一关系中各行、各列的顺序是任意的；每一列必须具有相同的数据类型。

1.3.2 关系模型的完整性约束条件

关系模式是通过关系数据描述语言描述关系后所生成的关系框架。

作为关系 DBMS，为了维护数据库的完整性，一般对关系模式提供了以下三类完整性约束机制。

1. 实体完整性约束规则

指关系中的“主键”不允许取“空值”(Null)。因为关系中的每一条记录都代表一个实体，而任何实体都是可标识的，如果主键值为空，就意味着存在不可标识的实体。

2. 引用完整性约束规则

亦称参照完整性约束规则。对一个关系上的外键，其值只允许两种可能：一是空值；二是等于外键对应的关系的主键值。这是由于不同关系之间的联系是通过“外键”实现的，当一个关系通过外键引用另一关系中的记录时，它必须能在引用的关系中找到这个记录，否则无法实现联系。

3. 用户定义的完整性约束

这是针对某一应用环境的完整性约束条件，它反映了某一具体应用所涉及的数据应满足的要求，往往是对关系模式中的数据类型、长度、取值范围的约束。DBMS 提供定义和检验这类完整性规则的机制，其目的是用统一的方式由系统来处理它们，不再由应用程序来完成这项工作。

1.3.3 关系数据操纵

关系型数据库管理系统提供关系操纵语言来实现数据操纵。

1. 关系数据操纵语言的特点

关系数据语言建立在关系代数基础上,具有以下特点:

(1)以关系为单位进行数据操作,操作的结果也是关系。

(2)非过程性强,很多操作只需指出做什么,而不需引导怎么做。

(3)以关系代数为基础,借助于传统的集合运算和专门的关系运算,使关系数据语言具有很强的数据操纵能力。

2. 关系代数

关系代数通过对关系的运算来表达查询。关系代数的运算包括传统的集合操作和专门的关系操作两类。

1) 传统的集合操作

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,即是对关系的行来进行的。

设关系 R 和关系 S 具有相同数目的属性列 (n 列属性) 并且相应的属性取自同一域,则可定义以下四种集合运算:

(1)并(Union): 关系 R 与关系 S 的并,记为 $R \cup S$ 。它是属于 R 或属于 S 的元组组成的集合,结果为 n 列属性的关系。

(2)交(Intersection): 关系 R 与关系 S 的交,记为 $R \cap S$ 。它是既属于 R 又属于 S 的元组组成的集合,结果为 n 列属性的关系。

(3)差(Difference): 关系 R 与关系 S 的差,记为 $R - S$ 。它是属于 R 而不属于 S 的元组组成的集合,结果为 n 列属性的关系。

(4)广义笛卡儿积(Extended Cartesian Product): 关系 R (假设为 n 列) 和关系 S (假设为 m 列) 的广义笛卡儿积,是一个 $(m+n)$ 列元组的集合,每一个元组的前 n 列是来自 R 的一个元组,后 m 列是来自关系 S 的一个元组。若 R 有 k_1 个元组, S 有 k_2 个元组,则关系 R 和关系 S 的广义笛卡儿积有 $k_1 * k_2$ 个元组。

2) 专门的关系运算

关系数据模型的理论基础是集合论,从集合的角度讲,可以进行集合的并、交、差运算;从关系(表)的角度讲,关系的基本运算有选择、投影和连接。

(1)选择: 选择运算是指从关系表中查找出符合指定条件的元组。它是对表的一种横向操作,其运算结果是关系中的部分元组并构成关系的一个子集,其关系模式不变。比如,从课程表中选取学时数是 80 的记录。

(2)投影: 投影运算是指从关系表中选取若干个属性,从而形成一个新的关系表。新关系表是原关系表的子集,投影是对表的一种纵向操作。例如,从课程表中对课程号、课程名、学分 3 个属性进行投影。

(3)连接: 连接运算是指从两个关系表中找出满足连接条件的所有元组,并且拼接成一个新的关系表。在 Visual FoxPro 中,连接运算是通过 JOIN 和 SELECT 等 SQL 命令来实现的。

第2章 Visual FoxPro 基础知识

2.1 Visual FoxPro 简介

Visual FoxPro 简称 VFP，是 Microsoft 公司推出的数据库开发软件，用它来开发数据库，既简单又方便。Visual FoxPro 源于美国 Fox Software 公司推出的数据库产品 FoxBase，在 DOS 上运行，与 xBase 系列相容。FoxPro 原来是 FoxBase 的加强版，最高版本曾出过 2.6。Fox Software 被微软收购后，可以在 Windows 上运行，并且更名为 Visual FoxPro。目前最新版本为 Visual FoxPro 9.0，而在学校教学和教育部门考证中还依然用经典版的 Visual FoxPro 6.0，它不仅提供了很多好的设计器、向导、生成器及新类，并且使得客户/服务器结构数据库应用程序的设计更加方便简捷。VFP6 以其强健的工具和面向对象的以数据为中心的语言，将客户/服务器和网络功能集成于现代化的、多链接的应用程序中，充分发挥了面向对象编程技术与事件驱动方式的优势。

2.1.1 Visual FoxPro 集成环境的使用

Visual FoxPro 启动后，屏幕上显示系统集成环境窗口，如图 2-1 所示。它主要由菜单栏、工具栏、状态栏、主窗口、命令窗口和项目管理器等部分组成。

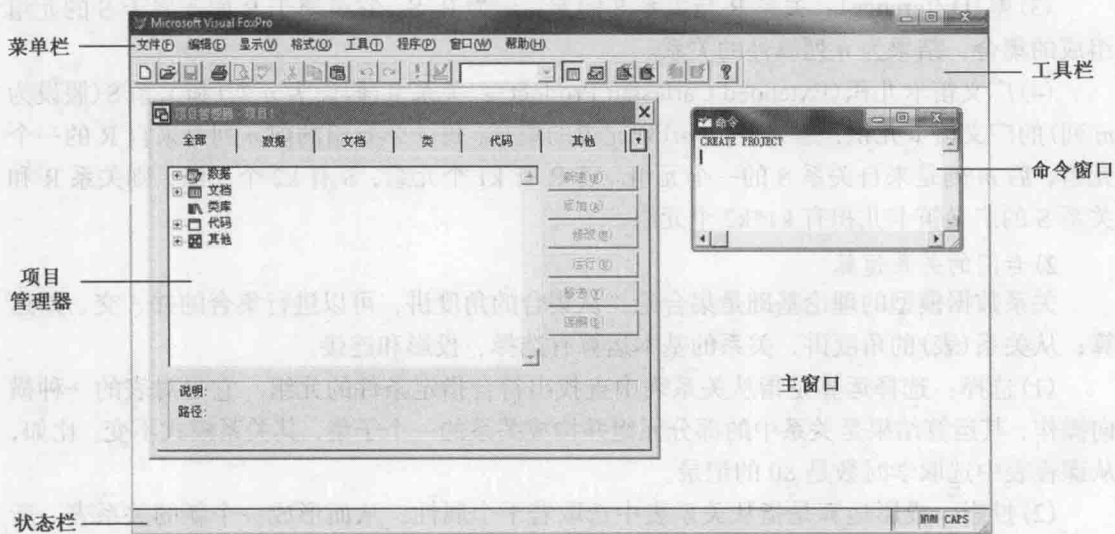


图 2-1 Visual FoxPro 集成环境