



工业和信息化普通高等教育“十二五”规划教材立项项目
21 世纪高等学校计算机规划教材
21st Century University Planned Textbooks of Computer Science

Visual FoxPro 程序设计教程

Visual Foxpro Programming

袁柱 游明英 主编

许莎 徐显秋 副主编

- 以全国计算机等级考试大纲为基础
- 结合多年教学经验和教改成果编写
- 实现理论与实践结合，教与学的统一



高校系列



人民邮电出版社
POSTS & TELECOM PRESS



工业和信息化普通高等教育“十二五”规划教材立项项目

21 世纪高等学校计算机规划教材

21st Century University Planned Textbooks of Computer Science

Visual FoxPro 程序设计教程

Visual Foxpro Programming

袁柱 游明英 主编

许莎 徐显秋 副主编



高校系列

人民邮电出版社

北京

图书在版编目 (C I P) 数据

Visual FoxPro 程序设计教程 / 袁柱, 游明英主编
— 北京: 人民邮电出版社, 2011. 2
21世纪高等学校计算机规划教材. 高校系列
ISBN 978-7-115-24743-8

I. ①V… II. ①袁… ②游… 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(2011)第010205号

内 容 提 要

本书是在《全国计算机等级考试(二级 Visual FoxPro 数据库程序设计)考试大纲》的基础上, 结合多年的教学经验和教改成果编写的。全书共 13 章, 内容包括数据库系统概述、Visual FoxPro 项目管理器、Visual FoxPro 的数据及其运算、表的操作、索引与统计、数据库的建立与使用、SQL 的应用、结构化程序设计、面向对象程序设计基础、表单设计与应用、菜单设计、报表与标签设计、数据库应用系统开发等。

本书适合作为高等院校“Visual FoxPro 程序设计”课程的教材使用, 也可以作为相关专业学生的参考书。

21 世纪高等学校计算机规划教材

Visual FoxPro 程序设计教程

-
- ◆ 主 编 袁 柱 游明英
副 主 编 许 莎 徐显秋
责任编辑 蒋 亮
 - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京鑫正大印刷有限公司印刷
 - ◆ 开本: 787×1092 1/16
印张: 18 2011 年 2 月第 1 版
字数: 472 千字 2011 年 2 月北京第 1 次印刷

ISBN 978-7-115-24743-8

定价: 32.00 元

读者服务热线: (010)67170985 印装质量热线: (010)67129223
防盗版热线: (010)67171154

[Handwritten notes on lined paper]

本书是在《全国计算机等级考试（二级 Visual FoxPro 数据库程序设计）考试大纲》的基础上、结合多年的教学经验和教改成果编写的。在详细介绍数据库管理系统设计基础知识、实际操作、数据库管理系统应用的基础上，强调理论与实践相结合，并注重实用性。全书共分为 13 章，每章都有知识点概述、小结、相关练习等配套内容，使读者能够在充分掌握 Visual FoxPro 数据库程序设计基础知识的同时，掌握数据库程序设计技术，从而实现教与学的结合统一。本书采用的例题大部分都是我们平时在教学过程中感觉针对性比较强，比较典型的题目，最后还特别增加了一个综合应用实例，把所学的主要内容综合起来，让读者基本掌握开发一个数据库应用系统的步骤和方法。全书通过大量应用实例介绍程序设计基础、面向对象程序设计的概念与方法，使读者轻松学会在 Windows 环境中的可视化编程。

本书由重庆科技学院计算机系长期从事计算机专业课和计算机基础课教学的教师编写的。其中第1章、第8章、第9章由游明英编写,第2章、第4章、第5章、第6章、第7章由许莎编写,第3章、第10章、第11章、第12章由袁柱编写,第13章由袁柱、徐显秋编写。袁柱完成了全书的统稿。

由于编者水平有限，书中难免会有错误和不足，敬请广大读者批评指正。

目 录

第 1 章 数据库系统概述..... 1	2.1.2 项目文件的打开..... 24
1.1 数据与数据处理..... 1	2.1.3 项目文件的关闭..... 25
1.1.1 数据和信息..... 1	2.2 项目管理器的界面..... 25
1.1.2 数据处理..... 2	2.2.1 项目管理器的选项卡..... 25
1.1.3 数据管理技术..... 2	2.2.2 项目管理器的命令按钮..... 26
1.2 数据库系统..... 3	2.2.3 定制项目管理器..... 27
1.3 数据模型..... 4	2.3 项目管理器的使用..... 28
1.3.1 数据模型概述..... 4	2.3.1 在项目管理器中新建或修改文件..... 28
1.3.2 实体及其联系..... 5	2.3.2 向项目中添加或移去文件..... 28
1.3.3 数据模型..... 6	2.3.3 项目文件的连编与运行..... 28
1.4 关系数据库..... 7	小结..... 29
1.4.1 关系模型..... 7	习题..... 29
1.4.2 关系数据库..... 8	第 3 章 Visual FoxPro 的数据及其运算..... 31
1.4.3 关系运算..... 9	3.1 Visual FoxPro 的数据类型..... 31
1.4.4 关系的完整性..... 10	3.2 Visual FoxPro 的常量与变量..... 32
1.5 Visual FoxPro 简介..... 11	3.2.1 常量..... 32
1.5.1 Visual FoxPro 的特点..... 11	3.2.2 变量..... 33
1.5.2 Visual FoxPro 文件的类型..... 12	3.3 Visual FoxPro 的内部函数..... 36
1.5.3 Visual FoxPro 命令格式..... 12	3.3.1 数值运算函数..... 36
1.6 Visual FoxPro 基本操作..... 13	3.3.2 字符处理函数..... 38
1.6.1 Visual FoxPro 启动、退出及用户界面..... 13	3.3.3 转换函数..... 39
1.6.2 Visual FoxPro 操作方式..... 16	3.3.4 日期和时间函数..... 41
1.7 Visual FoxPro 可视化设计工具..... 17	3.3.5 测试函数..... 41
1.7.1 Visual FoxPro 向导..... 17	3.4 Visual FoxPro 的基本运算..... 43
1.7.2 Visual FoxPro 设计器..... 18	小结..... 47
1.7.3 Visual FoxPro 生成器..... 19	习题..... 47
小结..... 19	第 4 章 表的操作..... 51
习题..... 20	4.1 表的建立..... 51
第 2 章 Visual FoxPro 项目管理器..... 23	4.1.1 设计表的结构..... 51
2.1 建立与打开项目文件..... 23	4.1.2 建立表的结构..... 53
2.1.1 项目文件的建立..... 23	4.1.3 向表输入记录..... 54

4.2 表的显示与维护	56	6.1.6 添加数据库表	94
4.2.1 表的打开与关闭	56	6.1.7 移去或删除表	97
4.2.2 表的显示	58	6.2 建立永久关系	97
4.2.3 表的修改	60	6.2.1 建立表之间的永久关系	97
4.2.4 表记录指针的定位	61	6.2.2 设置参照完整性	98
4.2.5 表记录的增加与删除	63	6.3 创建与使用查询	100
4.2.6 表的复制	66	6.3.1 查询的概念	100
小结	68	6.3.2 创建查询	100
习题	68	6.3.3 使用查询	109
第 5 章 索引与统计	72	6.4 视图的创建与使用	109
5.1 表的排序	72	6.4.1 视图简述	109
5.2 表的索引	73	6.4.2 创建视图	109
5.2.1 索引概述	73	6.4.3 使用视图	113
5.2.2 建立索引	74	小结	114
5.2.3 使用索引	76	习题	114
5.2.4 删除索引	79	第 7 章 SQL 的应用	118
5.3 查询	80	7.1 SQL 概述	118
5.3.1 顺序查询	80	7.2 SQL 的数据定义	119
5.3.2 索引查询	81	7.2.1 创建数据表	119
5.4 表的统计与计算	82	7.2.2 修改表的结构	123
5.4.1 计数命令	82	7.2.3 删除数据表	124
5.4.2 求和命令	82	7.2.4 定义/删除视图	124
5.4.3 求平均值命令	83	7.3 SQL 的数据查询	125
5.4.4 计算命令	83	7.3.1 SELECT 语句的格式	125
5.4.5 汇总命令	83	7.3.2 投影查询	126
5.5 多个表的同时使用	84	7.3.3 条件查询	127
5.5.1 使用工作区	84	7.3.4 统计查询	129
5.5.2 建立表间临时关系	85	7.3.5 分组查询	130
5.5.3 表的连接	87	7.3.6 查询的排序	130
小结	88	7.3.7 内连接查询	131
习题	88	7.3.8 自连接查询	132
第 6 章 数据库的建立与使用	92	7.3.9 修改查询去向	132
6.1 数据库的建立与管理	92	7.3.10 嵌套查询	133
6.1.1 建立数据库	92	7.3.11 集合查询	134
6.1.2 打开数据库	93	7.4 SQL 的数据操纵功能	135
6.1.3 修改数据库	94	7.4.1 插入记录	135
6.1.4 关闭数据库	94	7.4.2 更新记录	135
6.1.5 删除数据库	94	7.4.3 删除记录	136
		小结	136

习题	136	小结	186
习题	186		
第 8 章 结构化程序设计	140	第 10 章 表单设计与应用	190
8.1 程序设计概述	140	10.1 表单基础知识	190
8.1.1 程序设计的概念	140	10.1.1 表单概述	190
8.1.2 结构化程序设计方法	141	10.1.2 用表单向导建立表单	194
8.1.3 Visual FoxPro 程序的语法成分	142	10.1.3 用“表单设计器”建立表单	197
8.1.4 程序的书写规则	143	10.1.4 执行表单	204
8.2 程序文件的基本操作	143	10.2 表单控件的应用	204
8.2.1 程序文件的建立和编辑	143	10.2.1 控件操作概述	204
8.2.2 程序文件的运行	144	10.2.2 “标签”控件	205
8.3 常用命令的使用	144	10.2.3 “文本框”控件	205
8.3.1 输入命令	144	10.2.4 “命令”按钮控件	205
8.3.2 输出命令	146	10.2.5 “列表框”控件	206
8.3.3 其他常用命令	148	10.2.6 “组合框”控件	206
8.4 程序的控制结构	148	10.2.7 “编辑框”控件	206
8.4.1 顺序结构	148	10.2.8 “复选框”控件	207
8.4.2 分支结构	149	10.2.9 “选项按钮组”控件	207
8.4.3 循环结构	152	10.2.10 “微调按钮”控件	209
8.5 模块化程序设计	156	10.2.11 “计时器”控件	209
8.5.1 子程序	156	10.2.12 “图像”控件	209
8.5.2 自定义函数	157	10.2.13 “表格”控件	210
8.5.3 过程与过程文件	159	10.2.14 “页框”控件	210
8.5.4 内存变量的作用域和参数 传递	160	10.2.15 “命令按钮组”控件	211
小结	162	10.2.16 “ActiveX”控件和“ActiveX 绑定” 控件	211
习题	162	10.2.17 “表单集”控件	211
		小结	211
		习题	211
第 9 章 面向对象程序设计基础	171	第 11 章 菜单设计	215
9.1 面向对象程序设计基础	171	11.1 菜单设计概述	215
9.1.1 对象与类	171	11.1.1 菜单的基础知识	215
9.1.2 事件与方法	172	11.1.2 菜单设计的一般步骤	216
9.2 Visual FoxPro 中的类与对象	172	11.1.3 “菜单设计器”介绍	217
9.2.1 Visual FoxPro 的基类	172	11.1.4 为顶层表单添加菜单	219
9.2.2 对象的引用	173	11.1.5 设计快捷菜单	220
9.2.3 对象的事件与方法	176	小结	222
9.3 面向对象程序设计方法	178	习题	222
9.3.1 类的设计	178		
9.3.2 对象的设计	183		
9.3.3 调用方法	184		

第 12 章 报表与标签设计224

12.1 报表的基本概念224

12.1.1 报表的类型224

12.1.2 报表的数据源224

12.1.3 创建报表的方法225

12.2 创建报表225

12.2.1 使用“报表向导”创建报表225

12.2.2 使用“报表设计器”创建报表229

12.3 修改报表231

12.4 标签设计232

12.4.1 基本概念232

12.4.2 创建标签的方法232

小结234

习题235

第 13 章 数据库应用系统开发237

13.1 系统分析237

13.2 工资管理系统主要模块238

13.2.1 工资管理系统的主要功能238

13.2.2 工资管理系统功能模块图238

13.2.3 工资管理系统主要文件组成239

13.2.4 功能模块菜单239

13.3 项目与数据库的建立240

13.3.1 项目的建立240

13.3.2 数据库及数据表的建立240

13.4 工资管理系统模块设计241

13.4.1 系统封面模块表单设计241

13.4.2 系统主菜单设计243

13.4.3 工资录入模块表单设计245

13.4.4 工资浏览模块表单设计247

13.4.5 工资修改模块表单设计248

13.4.8 工资打印表单设计251

13.5 程序的连编252

13.5.1 将全部的应用程序添加到项目
管理器中252

13.5.2 主程序的设计253

13.5.3 在项目管理器中运行应用程序254

13.5.4 程序的连编254

小结255

习题255

附录 1 Visual FoxPro 常用文件

类型一览表256

附录 2 Visual FoxPro 6.0 常用命令

一览表257

附录 3 Visual FoxPro 6.0 常用函数

一览表269

参考文献280

第 1 章

数据库系统概述

本章主要内容有：信息、数据与数据处理的基本概念；数据模型相关知识；数据库管理系统与数据库应用系统；关系数据库相关知识；Visual FoxPro 简介和基本操作；Visual FoxPro 可视化设计工具的使用等。这些是学习和掌握 Visual FoxPro 技术的基础和前提。

通过本章的学习，读者应了解数据库管理系统相关的一些基础理论知识，掌握 Visual FoxPro 的基本操作和 Visual FoxPro 可视化设计工具的使用。

1.1 数据与数据处理

1.1.1 数据和信息

数据（Data）是人们用于记录事物情况的物理符号。为了描述客观事物而用到的数字、文字、图像、声音以及所有能输入到计算机中能被计算机处理的符号都可以看作数据。例如，一个人的年龄是 32 岁，一个教师的职称是“教授”，一个人的出生日期是“1972 年 10 月 1 日”，这里的 32、“教授”和“1972 年 10 月 1 日”等就是数据。

信息（Information）是客观事物属性的反映，它是某一客观事物中，某一事物的某一方面属性或某一时刻的表现形式。通俗地讲，信息是经过加工处理并对人类社会实践和生产活动产生决策影响的数据。不经过加工处理的数据只是一种原始材料，对人类活动产生不了决策影响，它只能起到记录客观世界的作用，数据经过加工处理后具有知识性才可以对人类活动产生有意义的决策作用。

数据与信息既有区别，又有联系。数据是表示信息的，但并非任何数据都能表示信息，信息只是加工处理后的数据，是数据所表达的内容。另一方面信息不随表示它的数据形式而改变，它是反映客观事物属性的，而数据则具有任意性，因不同的数据形式可以表示不同的信息。

例如，要描述教师人事档案时，人们感兴趣的可能是教师姓名、性别、出生日期、籍贯、政治面貌、职称等信息，对这些信息可以这样来描述：（李平，男，1971，四川，党员，教授），这就是关于李平教师的人事档案信息，它是通过一组数据反映出来的。

对于上面这组数据，了解的人会知道其具体含义，但是，不知道的人无法理解。因此，数据应该由数据值本身及其含义（型）两部分组成。当数据脱离其具体语义，则失去了意义。因此，数据和关于数据的解释是不可分的。换句话说，数据是有结构的，由型和值两部分组成。型表示

值的含义，或者表示值的语义。比如，数值“1971”，可以表示人的出生年份，也可以表示工资是1971等等，这个数值脱离了其语义，就变得没有意义了。

1.1.2 数据处理

所谓数据处理，实际上就是利用计算机将各种类型的数据转换成信息的过程。数据处理也叫信息处理。它包括对数据的收集、整理、存储、分类、排序、检索、维护、加工、统计、传输等一系列操作。数据处理的目的是从大量的、原始的数据中获得所需要的资料并提取有用的数据成份、作为行为或决策的依据。

1.1.3 数据管理技术

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，而数据库管理技术的发展，又使数据处理跨入了一个崭新的阶段。数据的管理技术的发展经历了人工管理、文件管理和数据库管理3个阶段。

（1）人工管理阶段（20世纪50年代中期以前）

人工管理出现在计算机应用于数据管理的初期，大约20世纪50年代中期以前。由于没有必要的软件和相应的硬件环境支持，用户只能直接在裸机上操作。在这个阶段数据是面向程序的，一个程序对应一组数据，由程序设计人员自己决定数据的组织、存储、输入、输出等，计算机基本上被当作一种计算工具。在这一管理方式下，用户的应用程序与数据不可分割，当数据有所变动时，程序则随之改变，独立性差，另外，各程序之间的数据不能相互传递，缺少共享性，因而这种管理方式既不灵活，也不安全，编程效率低下。

（2）文件系统阶段（20世纪50年代后期至60年代中期）

20世纪50年代后期至60年代后期，计算机开始大量用于数据管理。计算机硬件出现了直接存取的大容量外存储器，软件方面，出现了操作系统，包括文件系统。于是数据管理技术也进入到了文件管理阶段。所谓文件管理是利用文件系统管理软件，把相关的数据组织成一个数据文件，并长期地保存在外存储器上，这种数据文件可以脱离程序而独立存在，由一个专门的文件管理系统实现统一管理。在这一管理方式下，应用程序通过文件管理系统对数据文件中的数据进行加工处理。应用程序与数据之间具有一定的独立性。但数据文件仍高度依赖于与其对应的特定程序，不能被多个程序通用，由于数据文件之间不能建立联系，因而数据的通用性仍然较差，冗余量大。

（3）数据库系统阶段（20世纪60年代以来）

20世纪60年代以后，数据管理在文件系统的基础上发展到了数据库系统。所谓数据库管理就是通过数据库管理系统软件对所用的数据实行统一规划管理的，形成一个数据中心，构成一个数据“仓库”，在这个数据库中的数据能够满足不同用户的要求，供不同用户共享。在这种管理方式下，应用程序不再只与一个孤立的数据文件相对应，可以取整个数据集的某个子集作为逻辑文件与其对应，通过数据库管理系统实现逻辑文件与物理数据之间的协调与互动，从而实现数据处理。在数据库管理系统的系统环境下，应用程序对数据的管理和访问灵活方便，且数据与应用程序之间完全独立，程序的编制质量和效率都有所提高。由于数据库文件中的各数据子集间可以建立关联关系，所以数据的冗余大大减少，数据共享性显著增强。数据库技术的发展先后经历了层次数据库、网状数据库和关系数据库等几个阶段。层次数据库和网状数据库可以看作是第一代数据库系统，关系数据库可以看作是第二代数据库系统。自20世纪70年代提出关系数据模型和关系数据库后，数据库技术得到了蓬勃发展，应用也越来越广泛。

1.2 数据库系统

数据库系统实际上就是一个应用系统，也称数据应用系统。是把有关计算机硬件、软件、数据和有关人员组合起来为用户提供信息服务的系统。因此，数据库应用系统由计算机系统、数据库、数据库管理系统、相关人员和数据库应用程序组成。

1. 数据库

数据库 (Database) 是在数据库管理系统的集中控制之下，按一定的组织方式存储在计算机外存储器上的一组相关数据的集合。这些数据能够被多个用户共享，且具有冗余度小、独立性、保密性能好和安全性高等特点。

通俗地讲，数据库就是有条理、有组织、合理地存放数据的“数据仓库”。

数据库的性质由数据模型决定的，在数据库中，如果依照层次模型进行数据存储，则该数据库为层次数据库；如果依照网络模型进行数据存储，则该数据库为网络数据库；如果依照关系数据模型进行数据存储，则该数据库为关系数据库。

2. 数据库管理系统

数据库管理系统 (Database Management System, DBMS) 是数据库系统的核心软件之一，它提供数据定义、数据操作、数据库管理和控制功能。数据库管理系统通过对数据库进行统一的管理和控制，来保证数据库的安全性和完整性。用户通过数据库管理系统访问数据库中的数据，数据库管理员也通过数据库管理系统进行数据库的维护工作。数据库管理系统提供了多种功能，可使多个应用程序和用户用不同的方法在同时或不同时刻去建立，修改和询问数据库。数据库管理系统功能的强弱随系统而异，大系统功能较强、较全，小系统功能较弱。目前比较流行的数据库管理系统有 Oracle、SQL Server、Visual FoxPro、Access 等。

一般来说，数据库管理系统应该具有下列功能：

(1) 数据定义功能

DBMS 能向用户提供“数据定义语言” (Data Definition Language, 简称 DDL)，用于描述数据库的结构。

(2) 数据操作功能

对数据进行检索和查询，是数据库的主要应用。为此，DBMS 向用户提供“数据操作语言” (Data Manipulation Language, 简称 DML)，支持用户对数据库中的数据进行查询、更新) 包括增加、删除、修改) 等操作。

(3) 控制和管理功能

除了 DDL 和 DML 两类语句外，DBMS 还具有必要的控制和管理功能，其中包括：在多用户使用时对数据进行的“并发控制”；对用户权限实施监督的“安全性检查”；数据的备份、恢复和转储功能等。

3. 应用程序

应用程序是由用户编写的用来对数据库中的数据进行处理的程序。程序可用各种高级语言编写，数据库管理系统都提供了编写应用程序的语言，具有与各种高级语言相近的命令集，用户可以用这些命令编写应用程序。

4. 相关人员

数据库应用系统的有关人员主要有 3 类：最终用户、数据库应用系统开发人员和数据库管理员。最终用户指通过应用系统的用户界面使用数据库的人员。数据库应用系统开发人员包括系统分析员、系统设计员和程序员。数据库管理员是数据管理机构负责对整个数据库系统进行总体控制和维护，以保证数据库系统正常运行的一组人员。

存储于数据库中的大量数据是面向数据库结构的，数据库系统为数据的完整性、唯一性和安全性提供了一套统一且有效的管理手段，同时还提供了管理和控制数据的各种简单明了的操作命令，使用户程序编写简单、修改容易。

数据库系统具有以下主要特征：

（1）数据的独立性

在数据库系统中，数据库管理系统把数据与应用程序隔离开来，使数据独立于应用程序，当数据的存储方式和逻辑结构发生改变时，并不需要改变用户的应用程序。

（2）数据的共享性

存储在数据库中的数据能进行多种组合，以最优方式满足不同用户的需求。不同的用户可以使用数据库中的不同数据，也可以调用相同的数据。数据共享可以提高数据的利用率，减少数据的冗余度，有利于保持数据的一致性。

（3）可修改与可扩充性

数据库系统在结构和组织技术上是易于修改和扩充的。由于用户需求的不断变化，数据也需要不断扩充，数据库是逐步建立和完善起来的。

（4）统一的管理与控制

数据库系统能对数据进行必要的完整性管理与控制，以确保数据的正确、有效。

（5）安全与保密性

数据库系统应提供安全性与保密性措施，以使数据不会遭到破坏与盗用。

1.3 数 据 模 型

1.3.1 数据模型概述

在数据处理中，将涉及不同的数据描述领域。数据从反映事物的特征到计算机的具体表示，经历了 3 个领域——现实世界、信息世界和机器世界。

存在人们大脑之外的客观世界，称为现实世界。我们从现实世界中抽取数据库技术所研究的数据，进行分门别类，综合出系统所要的数据。

信息世界是现实世界在人们头脑中的反映，人们把它用文字和符号记载下来。信息世界的信息在机器世界中以数据形式存储。

为了把现实世界中的具体事物抽象、组织为某一 DBMS 支持的数据模型，人们常常首先将现实世界抽象为信息世界，然后将信息世界转换为机器世界。

为了进一步管理和控制数据，我们对现实世界数据特征进行抽象，并用一种模型表示出来，这种模型就是数据模型（Data Model）。

数据模型是反映客观事物及其联系的数据组织结构和形式。为了更准确反映现实世界，建立

数据模型时应满足三个方面的要求：一是比较能够真实地模拟现实世界；二是容易为人所理解；三是便于在计算机上实现。在数据库系统中，应针对不同的对象和应用，采用不同的数据模型。根据所处阶段不同，数据模型可以分为两类：一类是面向应用的概念模型，也称信息模型，它是按照用户的观点来对数据和信息建模，主要用于数据库设计阶段；另一类是面向数据库系统的数据模型，或称为基本数据模型，或称为逻辑数据模型，它主要用于数据库的实现，主要包括层次模型、网状模型、关系模型等。

1.3.2 实体及其联系

现实世界中的客观事物千姿百态，各种不同类型的事物很容易区别开来。同一类事物，如两个人，可以通过一些特征如姓名、性别、年龄、身高等来加以区别。同时，信息世界中的事物总是息息相关的，如学生与教师之间、学生与课程之间的联系等。要将这些事物以数据的形式存储在计算机中，人们必须经历对现实世界中事物特征的认识、然后抽象概念化，最后组织成计算机数据库等过程，即把现实世界转化为信息世界。

(1) 实体 (Entity)

客观存在并可相互区别的事物在信息世界中称为实体。这些事物既可以是具体的人、事、物（如一个学生、一本书等），也可以是抽象的概念或关系（如一门课程、一场考试等），这些都是实体。

(2) 属性 (Attribute)

实体所具有的特性在信息世界中称为属性。一个实体可由若干属性来表示。如某个学生实体可由学号、姓名、性别、年龄、专业等属性组成，这些属性的具体值就是对某个学生的描述。换句话说，即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。

(3) 实体集 (Entity Set)

实体集是具有相同特性的实体的集合。如在一个学校，所有的老师组成一个教师实体集，所有的学生组成了一个学生实体集，所有的课程组成了一个课程实体集。

(4) 码 (Key)

能唯一标识实体集中每个实体的属性或属性集，称为实体的码（或键）。例如，学号是学生实体的码。

(5) 联系 (Relationship)

实体之间相互的关联关系，称为联系，如教师与学生之间的联系就是教师教学生。实体之间的联系是各种各样的。两个不同实体集的实体间联系有以下3种情况。

① 一对一的联系 (1:1)：如果对于实体集 A 中的每一个实体，实体 B 中有且只有一个实体与之联系，反之亦然，则实体 A 与实体 B 具有一对一联系。例如：一个学校只有一位校长，反之，一个校长只能在一所学校任职。所以学校与校长的联系就是一对一的联系。如图 1-1 (a) 所示。

② 一对多的联系 (1:m)：如果对于实体集 A 中的每一个实体，实体 B 中有多个实体与之联系，反之，对于实体集 B 中的每一个实体，实体 A 中有且只有一个实体与之联系，则实体 A 与实体 B 具有一对多的联系。如一个班级有许多学生，但一个学生只能在一个班上就读，所以班级和学生之间的联系就是一对多的联系。如图 1-1 (b) 所示。

③ 多对多的联系 (m:n)：如果对于实体集 A 中的每一个实体，实体 B 中有多个实体与之联系，反之，对于实体集 B 中的每一个实体，实体 A 中也有多个实体与之联系，则实体 A 与实体 B 具有多对多的联系。如一名学生可以选修若干课程，而一门课程可以被多名学生选修，所以学生

和课程之间的联系就是多对多的联系。如图 1-1 (c) 所示。

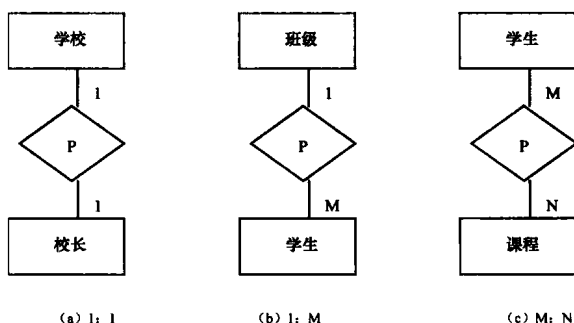


图 1-1 两个实体型间联系

(6) 值域 (range)

值域是实体属性取值的范围。如成绩一般在 0~100 之间, 性别的取值只能是“男”或“女”。这种属性的取值范围称为值域。

1.3.3 数据模型

概念模型的表示方法很多, 其中最为著名的是 1979 年 P.P.S.Chen 提出的实体-联系模型 (Entity-Relationship Model, 简称 E-R 型), 这个模型是直接从现实世界中抽象出来实体类型及实体间联系, 然后用 R-R 图表示的模型。在 E-R 图中, 用方框表示实体型, 用菱形表示实体间的联系, 用椭圆表示实体或联系的属性, 实体间联系用箭头标出并注上联系的种类。

面向数据库系统常用的数据模型包括层次模型、网状模型和关系模型等 3 种。

(1) 层次模型

层次模型用树形结构来表示实体以及它们之间的联系。如图 1-2 所示, 每个结点是一个实体, 每一条线表示两个实体类型之间的关系。这种关系又称作亲子关系。例如, 在图 1-2 中一个院系有多个专业, “学校”称为双亲, “院系”称为子女。

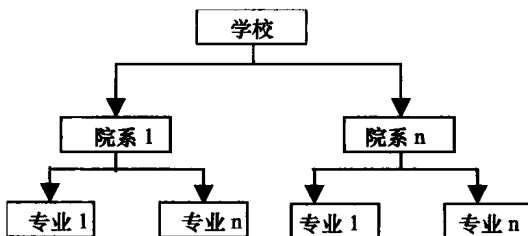


图 1-2 数据的层次模型

层次模型的特点如下:

- ① 有且只有一个结点没有父结点, 这个结点称为根结点。
- ② 除根结点外, 其余结点有且仅有一个父结点。

(2) 网状模型

网状模型是层次模型的扩展, 其总体结构呈现一种交叉的网状结构, 在两个数据之间允许存在两种或多种的联系。网状模型的示例如图 1-3 所示。其主要特征如下。

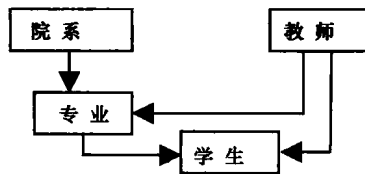


图 1-3 数据的网状模型

- ① 有一个以上结点无双亲；
- ② 允许结点有一个以上双亲。

(3) 关系模型

关系模型是用人们熟悉的二维表来组织和存储数据,也是目前应用比较广泛的一种数据模型。关系模型的示例如表 1.1 所示。关系模型的详细介绍见 1.4 小节。

表 1.1 数据的关系模型

编号	姓 名	性别	出生日期	党员否	专 业	职 称	英语水平
100101	张东伟	男	1968. 08	T	计算机应用	教授	精通
201002	黄中华	男	1972.10	T	计算机网络	副教授	精通
100102	李小菊	女	1976. 03	F	计算机应用	讲师	精通

1.4 关系数据库

1.4.1 关系模型

1. 关系模型中的基本概念

(1) 字段

关系模型是用人们熟悉的二维表来组织和存储数据。信息世界中的“属性”,就是数据世界中的“数据项”,从数据库的角度讲,数据项就是字段;从表格的角度,数据项称为列。由此可见,字段、属性、数据项、列这些术语,所描述的对象是相同的。如表 1.2 教师信息表每条记录就由“编号”、“姓名”、“性别”、“出生日期”、“党员否”、“专业”、“职称”、“英语水平”等字段构成。

(2) 记录

字段的有序集合称为记录。在关系模型中,记录称为元组;在表中,记录称为行;在信息世界中称为实体。换句话说,实体、记录、元组和行分别是不同角度上描述同一对象的术语。在表 1.2 信息表中,每一行就是一个字段或一个元组。

(3) 关系

一个关系就是一张二维表,通常将一个没有重复行、重复列的二维表看成一个关系,每个关系都有一个关系名。在 Visual FoxPro 中,一个关系对应于一个表文件,它是记录的集合,其扩展名为.dbf。例如表 1.2 教师信息表就是一个关系。

(4) 关键字

关键字是可以将表中的各记录区分开的记录的属性。表中一定存在这样的属性或属性组合,它的值能够识别表中的每一条记录,该属性或属性的组合就称为该记录的关键字。如教师的编号属性就可以设为教师记录的关键字,因为不可能有两个教师编号是相同的。

① 候选关键字: 关系中能够成为关键字的属性或属性组合可能不是唯一的。凡在关系中能够唯一区分、确定不同元素的属性或属性组合,称为候选关键字。

② 主关键字与外部关键字: 从所有候选键中选取一个作为用户使用的键称主键,也叫主关键字。表 A 中的某属性是某表 B 的键,则称该属性集为 A 的外键、外码或外部关键字。

2. 关系的基本性质

在关系模型中，关系具有如下性质：

① 关系必须规范化，是一张每个属性值都不可分割的二维表。

规范化是指关系模型中每个关系模式都必须满足一定条件的要求，最基本的要求是关系必须是一张每个属性值都不可分割的二维表。即表中不能再包含表。例如，表 1.2 就不能直接作为一个关系。因为该表“自然情况”、“专业”、等都可以分成若干子列。这与每个属性不可再分割的要求不符。只要去掉“自然情况”、“专业”等项，而将编号、姓名、专业、职称等直接作为基本的数据项就可以了。

表 1.2

教师基本信息情况一览表

自然情况					专 业		
编 号	姓 名	性 别	出生日期	党员否	专 业	职 称	英语水平
100101	张东伟	男	1968. 08	T	计算机应用	教授	精通
201002	黄中华	男	1972.10	T	计算机网络	副教授	精通
100102	李小菊	女	1976. 03	F	计算机应用	讲师	精通
...		...		..	..		...

② 在同一关系中不允许有重复的属性名。

③ 同一关系中不允许有完全相同的记录，且记录的顺序可以是任意的。

④ 在同一关系中同一列的数据类型必须相同，且各列顺序可以是任意。

以上是关系的基本性质，也是衡量一个二维表格是否构成关系的基本要素。在这些基本要点中，有一点是关键，即属性不可分割，即表中不能嵌套。

1.4.2 关系数据库

关系数据库是若干个依照关系模型设计的相关关系的集合。也就是说，关系数据库通常由若干个有一定关系的二维表组成的，其中至少有一个表。

一个“关系”就是一张二维表格，称为一个数据表文件（简称数据表）。数据表是由数据及表结构构成。表结构对应关系模型，表格每一列对应关系模型的属性，该列的数据类型和取值范围就是属性的域。因此，定义了表结构就定义了对应的关系。

在 Visual FoxPro 中，与关系数据库对应的是数据库文件（.dbc 文件），一个关系数据库由若干个数据表组成，一个数据表又由若干数据记录组成，而每一个记录则由若干个以字段属性加以分类的数据项组成。

在关系数据库中，每一个数据表都具有相对的独立性，这一独立性的唯一标志是数据表的名字，称为表文件（.dbf）。一个数据库中不允许有重复的数据表，因为对数据表中数据的访问是通过表文件名来导引的。

在关系数据库中，有些数据表之间具有相关性，数据表之间的这种相关性是依靠每一个独立的数据表内部具有相同属性的字段建立的。

下面看一个在学生管理系统中的学生管理数据库的实际例子，该数据库中有“学生信息表”、“选课表”、“课程表”等，如图 1-4 所示。由于每个学生可以选若干课程，一门课程也可以被多个学生选修，所以学生和课程之间的联系是多对多的联系，通过选课表可以把多对多的关系分解为两个一对多的关系，所以建立的“学生信息”-“选课”-“课程”之间的关系如

图 1-5 所示。“学生信息表”和“选课表”之间的相关性依靠的是“学号”字段建立，“选课”表与“课程”之间依靠的是“课程号”字段建立的相关性。

学生信息表

学号	姓名	出生日期	少数民族	籍贯	入学成绩	简历	照片
001	张 杰	05/12/88	T	北京			
002	李 刚	04/30/87	F	天津			
003	王大同	12/03/88	T	山西			
004	李建中	10/25/87	T	陕西			
005	王永民	09/18/88	T	河南			
006	张 玲	01/06/88	T	湖北			
007	林 森	07/23/87	F	浙江			
008	曾小梅	08/28/88	F	湖南			
009	熊 敏	02/20/87	F	江西			
010	李文丽	06/15/88	T	山东			
011	刘 斌	12/13/87	F	山西			

选课

学号	课程号	成绩
001	340001	90
002	340002	68
001	340002	85
001	34003	92
005	340001	96

课程

课程号	课程名	学分
340001	课程名计算机导论	2
340002	课程名数据结构	4
340003	课程名计算机组成	3
340004	课程名计算机网络	4
340005	课程名数据库原理	4

图 1-4 学生管理数据库中的表

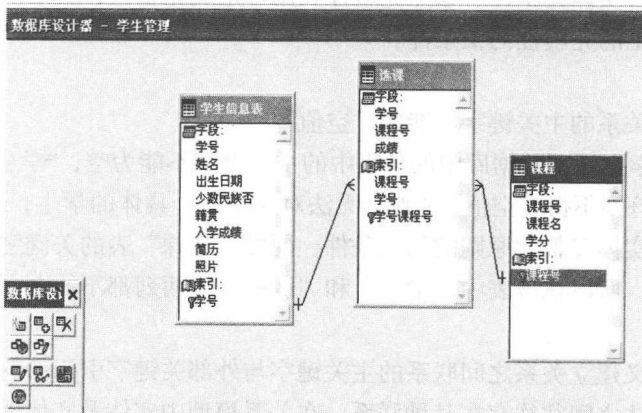


图 1-5 学生管理数据库中表之间的关系

1.4.3 关系运算

在关系数据库中查询用户所需要的数据时，需要对关系进行一定的关系运算。关系运算主要包括选择、投影和联接 3 种。

1. 选择

选择运算是从关系中查找符合指定条件元组的操作。在选择运算中，以逻辑表达式指定查询条件，选择运算将选取使逻辑表达式为真的所有元组。选择运算的结果构成关系的一个子集，是关系中的部分元组，其关系模式不变。

选择运算是从二维表格中选取符合条件的若干行的操作，在表中则是选取若干记录。在 Visual FoxPro 中，可以通过子句 FOR<逻辑表达式>、WHILE<逻辑表达式>和设置记录过滤器实现选择运算。

2. 投影

投影运算是从关系中选取若干属性的操作。投影运算从关系中选取若干属性形成一个新的关系，新关系的关系模型中属性个数比原关系少，或者排列顺序不同，同时也可能减少元组个数。