

※ 21世纪高校计算机教材

Visual FoxPro

程序设计



应用教程

主 编

匡 松 刘 容

主 审 黄迪明

副主编

易 涛 何振林

邱俊山 黄 敏



电子科技大学出版社

Visual FoxPro

程序设计

应用教程

主 编：匡 松 刘 容

主 审：黄迪明

副主编：易 涛 何振林

邱俊山 黄 敏

顾 问：王正智 黎 明 马义玲 唐向阳

孙淑霞 潘世永 吴家培

编 委：（以姓氏拼音字母为序）

陈 年 高仕忠 郭兴吉 李明龙 梁金明 刘 毅 刘成安

刘家彬 罗明英 孟 丽 孟 林 宋 健 王 超 王华军

王学平 肖 甘 徐 鹰 杨 凡 杨海中 杨清平 张 平

张庆荣 张选芳 赵 亮 周 舸 周 敏 周学文 朱福成

电子科技大学出版社

图书在版编目 (CIP) 数据

Visual FoxPro 程序设计应用教程/ 匡松, 刘容主编.

—成都: 电子科技大学出版社, 2004.1

ISBN 7-81094-410-X

I.V... II.匡... III.关系数据库—数据库管理系统, Visual FoxPro—水平考试—教材
IV.TP311.138

中国版本图书馆 CIP 数据核字 (2004) 第 003293 号

内 容 提 要

本教程按 Visual FoxPro 程序设计计算机等级 (二级) 考试大纲编写, 系统介绍了表的创建、排序和索引、数据库操作、SQL 语言、面向过程和面向对象的程序设计方法、表单设计、报表设计等内容; 并介绍了数据结构、软件工程、操作系统的软件基础内容。

本教程同时配套出版附有参考答案和重点解析的《Visual FoxPro 程序设计上机实验和考级训练教程》(匡松 何振林主编, 电子科技大学出版社, 2004.1)。

本套教材既适宜作为省级计算机等级考试教材, 也适宜作为全国计算机等级考试教材。

Visual FoxPro 程序设计应用教程

匡松 刘容 主编

出 版: 电子科技大学出版社 (成都市建设北路二段四号)

责任编辑: 周 岚

特约编辑: 孙康江

发 行: 电子科技大学出版社

印 刷: 电子科技大学出版社印刷厂

开 本: 850mm×1168mm 1/16 印张 17.5 字数 550 千字

版 次: 2004 年 2 月第一版

印 次: 2004 年 2 月第一次印刷

书 号: ISBN 7-81094-410-X/TP·244

印 数:

定 价: 28.00 元

目 录

第一章 数据库基本理论	1
1.1 数据、信息和数据处理	2
1.1.1 数据与信息	2
1.1.2 数据处理	2
1.2 数据管理技术的发展	2
1.2.1 人工管理阶段	3
1.2.2 文件管理阶段	3
1.2.3 数据库系统阶段	3
1.3 数据库系统基本概念	4
1.3.1 数据库	4
1.3.2 数据库管理系统	5
1.3.3 数据库系统	5
1.3.4 数据库应用系统	5
1.4 数据模型	5
1.4.1 层次模型	6
1.4.2 网状模型	6
1.4.3 关系模型	6
1.5 关系数据库基础知识	7
1.5.1 关系术语	7
1.5.2 关系运算	7
1.5.3 关系数据库	8
1.5.4 关系的完整性	10
第二章 Visual FoxPro 系统概述	12
2.1 Visual FoxPro 的特点及性能指标	13
2.1.1 Visual FoxPro 的特点	13
2.1.2 Visual FoxPro 的性能指标	14
2.2 Visual FoxPro 的启动与用户界面	14
2.2.1 Visual FoxPro 的运行环境	14
2.2.2 Visual FoxPro 的启动与退出	14
2.2.3 Visual FoxPro 的界面	15
2.2.4 Visual FoxPro 的菜单系统	16
2.2.5 工作区窗口和命令窗口	18
2.3 Visual FoxPro 的工作方式 和命令结构	18
2.3.1 工作方式	18

2.3.2 命令结构	19
2.4 Visual FoxPro 的辅助工具	19
2.4.1 向导	19
2.4.2 设计器	21
2.4.3 生成器	21
2.5 项目管理器	22
2.5.1 创建项目	22
2.5.2 项目管理器的使用	23
2.5.3 定制项目管理器	24
第三章 Visual FoxPro 的数据基础	25
3.1 数据类型	26
3.2 常量	27
3.3 变量	27
3.3.1 内存变量	27
3.3.2 数组变量	28
3.3.3 字段变量	29
3.3.4 系统变量	29
3.4 表达式	29
3.5 常用函数	31
3.5.1 数值运算函数	31
3.5.2 字符处理函数	33
3.5.3 转换函数	35
3.5.4 日期函数	36
3.5.5 测试函数	37
3.6 Visual FoxPro 的文件类型	38
3.7 Visual FoxPro 表的类型	39
第四章 表的建立和操作	40
4.1 表的建立	41
4.1.1 数据表的结构	41
4.1.2 表结构建立	42
4.1.3 表数据的输入	44
4.2 数据表的打开、显示和关闭	46
4.2.1 打开表	46
4.2.2 表文件的显示	46
4.2.3 关闭表	48
4.3 表的修改与编辑	49

4.3.1 表结构的修改	49
4.3.2 表的记录修改和编辑	49
4.3.3 数据表与数组之间的数据交换	54
4.4 数据表的过滤	55
4.4.1 记录过滤	55
4.4.2 字段过滤	56
4.5 数据表的复制	56
4.5.1 复制任何类型的文件	56
4.5.2 数据表结构的复制	57
第五章 排序、查询和多表操作	59
5.1 排 序	60
5.2 索 引	61
5.2.1 索引的概念	61
5.2.2 索引文件的建立	62
5.2.3 索引的使用、更新与删除	64
5.3 查 询	67
5.3.1 顺序查询	67
5.3.2 索引查询	68
5.4 统计与汇总	69
5.4.1 计数命令 COUNT	69
5.4.2 求和命令 SUM	69
5.4.3 求平均命令 AVERAGE	70
5.4.4 计算命令 CALCULATE	70
5.4.5 汇总命令 TOTAL	70
5.5 多工作区的操作	71
5.5.1 工作区的选择与数据工作期	71
5.5.2 表之间的关系	73
5.5.3 表之间的连接	75
第六章 数据库操作与视图的使用	77
6.1 数据库的创建与管理	78
6.1.1 数据库的创建	78
6.1.2 在项目中添加数据库	79
6.1.3 数据库的打开、修改和关闭	80
6.1.4 数据库对表的管理	81
6.1.5 数据库文件察看	83
6.2 数据字典	83
6.2.1 字段属性	83
6.2.2 记录规则	85
6.2.3 永久关系	86
6.2.4 设置参照完整性	87

6.3 视 图	88
6.3.1 本地视图的创建	88
6.3.2 远程视图的创建	93
第七章 结构化查询语言 SQL	96
7.1 SQL 的数据定义功能	97
7.1.1 SQL 提供的基本数据类型	97
7.1.2 创建、修改和删除数据表	97
7.1.3 创建和删除索引	99
7.2 SQL 的数据查询功能	100
7.2.1 SELECT 命令的格式	100
7.2.2 投影查询	101
7.2.3 条件查询	102
7.2.4 统计查询	104
7.2.5 分组查询	105
7.2.6 查询的排序	106
7.2.7 连接查询	106
7.2.8 嵌套查询	109
7.2.9 查询结果输出	111
7.3 SQL 的数据更新功能	112
7.3.1 插入数据记录	112
7.3.2 修改数据记录	112
7.3.3 删除数据记录	113
7.4 用查询设计器建立查询	113
7.4.1 进入查询设计器	114
7.4.2 创建查询	116
7.4.3 指定查询去向	116
7.4.4 查看 SQL	117
7.4.5 运行、保存和修改查询	117
7.4.6 查询菜单	117
第八章 Visual FoxPro 程序设计基础	118
8.1 面向过程程序设计	119
8.1.1 程序文件的建立和执行	119
8.1.2 程序中常用的命令	120
8.1.3 程序的基本结构	124
8.1.4 过程及过程调用	130
8.1.5 变量的作用域和参数调用	131
8.1.6 数组应用	133
8.2 面向对象程序设计	134
8.2.1 基本概念	134
8.2.2 面向对象程序设计基本方法	136
8.2.3 Visual FoxPro 中的类	136

8.2.4 属 性	137
8.2.5 事 件	138
8.2.6 方法程序	138
8.2.7 数据环境	138
8.2.8 对象的操作	138
8.2.9 设计类与对象	139
8.3 调试程序	143
8.3.1 调试器窗口	143
8.3.2 调试器应用	143
第九章 表单设计	145
9.1 表单设计基础	146
9.1.1 表单简介	146
9.1.2 表单向导	148
9.1.3 表单设计器	151
9.2 表单常用控件	155
9.2.1 控件的基本操作	155
9.2.2 标签控件	155
9.2.3 文本框控件	157
9.2.4 命令按钮控件	158
9.2.5 常用表单控件应用	159
9.3 表单其他控件	160
9.3.1 选项按钮组	160
9.3.2 复选框	161
9.3.3 编辑框	162
9.3.4 列表框	163
9.3.5 组合框	164
9.3.6 微调按钮	164
9.3.7 计时器	165
9.3.8 图 像	166
9.3.9 表 格	167
9.3.10 页 框	169
9.3.11 命令按钮组	170
9.3.12 ActiveX 控件和 ActiveX 绑定控件	173
9.3.13 表单集	174
9.3.14 表单控件的综合应用	174
第十章 Visual FoxPro 程序设计	182
综合应用	182
10.1 数据库操作实例	183
10.1.1 浏览数据实例	183
10.1.2 维护数据实例	184
10.1.3 学生选课实例	188

10.1.4 成绩查询实例	191
10.1.5 教师课程查询实例	193
10.1.6 计算平均年龄实例	194
10.2 表单控件应用实例	195
10.2.1 求素数实例	195
10.2.2 打字测试实例	197
10.2.3 算术计算实例	198
10.2.4 数值排序实例	200
第十一章 报表设计	202
11.1 报表设计基础	203
11.1.1 报表的常规布局	203
11.1.2 报表设计的步骤	203
11.1.3 创建报表布局文件	204
11.2 创建简单报表	204
11.2.1 创建空白报表	204
11.2.2 创建快速报表	204
11.2.3 用报表向导创建报表	205
11.3 报表设计器	207
11.3.1 报表设计器的基本环境	207
11.3.2 设置报表的数据环境	208
11.3.3 报表控件	209
11.3.4 报表控件的调整与控制	212
11.4 报表打印输出	212
11.4.1 页面设置	212
11.4.2 预览结果	213
11.4.3 打印报表	213
11.5 报表设计示例	214
第十二章 应用系统的集成	217
12.1 菜单系统	218
12.1.1 菜单系统的基本结构	218
12.1.2 菜单系统的设计步骤	219
12.2 设计菜单	220
12.2.1 菜单设计器的使用	220
12.2.2 主菜单中的“显示”和“菜单” 下拉菜单中有关选项	221
12.2.3 下拉式菜单设计示例	222
12.2.4 快捷菜单	225
12.3 应用系统的集成	225
12.3.1 应用系统启动主程序的设计	225
12.3.2 项目集成	227

12.3.3 应用程序的发布	228
第十三章 软件技术基础	230
13.1 数据结构	231
13.1.1 数据结构的基本概念	231
13.1.2 数据结构的意义	231
13.1.3 算法分析	232
13.1.4 线性表	233
13.1.5 树	237
13.1.6 图结构	239
13.1.7 线性表的查找	241
13.1.8 内排序	242
13.2 软件工程	244
13.2.1 概 述	244
13.2.2 软件生存周期	245

13.2.3 软件定义	246
13.2.4 软件开发	247
13.2.5 软件测试	248
13.2.6 软件维护	248
13.2.7 计算机辅助软件工程	249
13.3 操作系统	249
13.3.1 操作系统的概念和类型	249
13.3.2 处理机管理	250
13.3.3 存储器管理	253
13.3.4 设备管理	254
13.3.5 文件管理	255
附录一 Visual FoxPro 6.0 常用函数表	257
附录二 Visual FoxPro 6.0 命令概要	261
附录三 双语教学参考英语词汇	269

第一章

数据库基本理论

本章双语教学参考英语词汇

参照完整性	consult integrity	数据处理	data processing, data handing
层次模型	hierarchical model	数据管理	data management/administration
发展	development	数据库	database
分布式数据库	distributed database	数据库管理系统	database management system
概述	briefly, in broad outline	数据模型	data model
关键字	key word	数据冗余	data redundancy
关系	relation, relationship	术语	terms, terminology
关系模式	relation schema/scheme	属性	attribute, quality
关系模型	relation model, relational model	完整性	integrity
关系数据库	relational database	网状模型	network model
关系运算	relation(al) operation	文件管理	document management
基本知识	basic knowledge	系统	system
集中式数据库	centralized database	信息	information message
技术	technology	选择	select, choose
阶段	stage	应用系统	applied system
控制	control	域	territory, region, area, field, domain
连接	connection, link, join, bind, connect, conjunct	域完整性	territory integrity
人工管理	man-made management	元组	tuple
投影	project(ion)	运算	operate, operation
实体完整性	entity integrity, entity-integrity property		
数据	data		



本章主要介绍信息、数据、数据处理、数据模型的基本概念,着重介绍数据库、数据库管理系统、数据库系统的功能、组成和相互关系,以及关系数据库的三种关系运算和关系表中的记录、字段、关键字段、关系模型等概念。

1.1 数据、信息和数据处理

1.1.1 数据与信息

1. 数据

数据是对客观事物记录下来的事实,是描述或表达信息的物理形式。在计算机领域,凡能为计算机所接受和处理的物理形式,例如字符、数字、图形、图像、声音等都可称之为数据。因此,数据泛指一切可被计算机接受和处理的符号。数据可分为数值型数据(如产量、价格、成绩等)和非数值型数据(如人名、日期、文章、声音、图形、图像等)。数据可以被收集、存储、处理(加工、分类、计算等)、传播和使用。

2. 信息

信息是指数据经过加工处理后所得到的有价值的知识。信息无时不有,无处不在,客观存在于人类社会的各个领域,而且不断地变化着。我们需要不断地获取信息、加工信息和运用信息为社会的各个领域服务。从计算机应用的角度,我们通常将信息看做人们进行各种活动所需要获取的知识。

信息与数据既有联系又有区别,数据反映了信息,而信息又依靠数据来表达。用不同的数据形式可以表示同样的信息,信息不随它的数据形式不同而改变。例如:某个部门要召开会议,这个事件形成了“开会”这样一个信息。把这个信息通知有关单位时,可以使用广播,即通过“声音”这个具体形式;也可以通过文件,以文字形式向有关单位传递。“开会”这一信息就从两种不同数据得到。尽管数据形式不同,但“开会”这个信息的内容没有变,因此可以说信息是数据的内涵,而数据是信息的具体表示形式。在许多地方,信息和数据并不是截然分开的,因为有些信息本身就是数据化的,数据本身又是一种信息。因此,在多数情况下不对它们进行区分,计算机进行数据交换也可以说是信息交换,数据处理也意指信息处理。总之,信息是反映客观现实世界的知识,数据是信息的具体表现形式,数据经过加工处理后使其具有知识性并对人类活动产生有意义的决策作用。

1.1.2 数据处理

数据处理是指将数据转换成信息的过程。数据处理主要包括对数据的收集、分类、排序、存储、计算、加工、检索、传输和制表等处理过程,经过处理提炼的数据能够反映事物或过程的本质和内在联系,为我们工作和决策提供必要的信息基础和依据,输出有价值、有意义的信息。

1.2 数据管理技术的发展

数据库(Data Base, 简称为DB)技术是20世纪60年代末出现的以计算机技术为基础的数据处理技术。数据处理的核心问题是数据管理。数据管理指的是对数据进行组织、编码、分类、存储、检索与维护等操作。随着计算机硬件技术和软件技术的发展和进步,计算机数据管理的水平不断提高,管理方式也发生了很大的变化。发展到现在,数据管理经历了人工管理、文件管理和数据库系统(Data Base System, 简称为DBS)三个阶段。



1.2.1 人工管理阶段

人工管理阶段起始于 20 世纪 50 年代。当时计算机的存储设备没有磁盘，数据只能存放于卡片、纸带上。在软件方面，也没有专门的管理数据的文件，数据由计算数据的程序携带。在人工管理阶段对数据的管理存在的主要问题是：①数据不能独立，编写的程序是针对程序中携带的数据，当数据修改时程序也得修改。而程序修改后，数据的格式、类型也得变化以适应处理它的程序。②数据不能长期保存，数据被包含在程序中。程序运行结束后，数据和程序一起从内存中释放。③没有对数据管理的软件，即当时还没有开发专门进行数据管理的软件。人工管理阶段不仅要设计数据的处理方法，而且还要说明数据在存储器的存储地址。应用程序和数据是一一对应的，各程序之间的数据不能相互传递，数据不能重复使用。

1.2.2 文件管理阶段

在 20 世纪 60 年代，计算机软、硬件技术得到快速发展，硬件有了磁盘、磁鼓等大容量且能长期保存数据的存储设备，软件有了操作系统。操作系统中有专门的文件系统用于管理外部存储器上的数据文件，数据与程序分开，数据能长期保存。文件管理阶段可以把有关的数据组织成一个文件，这种数据文件可以脱离程序而独立存储在外存储器上，由一个专门的文件管理系统对其进行管理。与早期人工管理阶段相比，文件管理的效率和数量都有很大提高，但仍存在以下问题：

(1) 数据没有完全独立：虽然数据和程序分开，但所设计的数据是针对某一特定程序，所以无论是修改数据文件和程序文件都要相互影响。

(2) 存在数据冗余：文件系统中的数据没有合理、规范的结构，使得数据的共享性极差，哪怕是不同程序使用部分相同的数据，数据结构也有一点不同，都要创建各自的数据文件，造成数据的重复存储。

(3) 数据不能集中管理：文件系统中的数据文件没有集中的管理机制，数据的安全性和完整性都不能保障。各数据之间、数据文件之间缺乏联系，给数据处理造成不便。

1.2.3 数据库系统阶段

由于文件系统管理数据的缺陷，迫切需要新的数据管理方式，把数据组成合理结构，能集中、统一地进行管理。数据库技术开始于 20 世纪 60 年代末，在美国产生了具有商业价值的数据库系统，20 世纪 80 年代随着微型机的普遍应用和数据库系统的不断完善，数据库系统在世界范围内得到了广泛的应用。

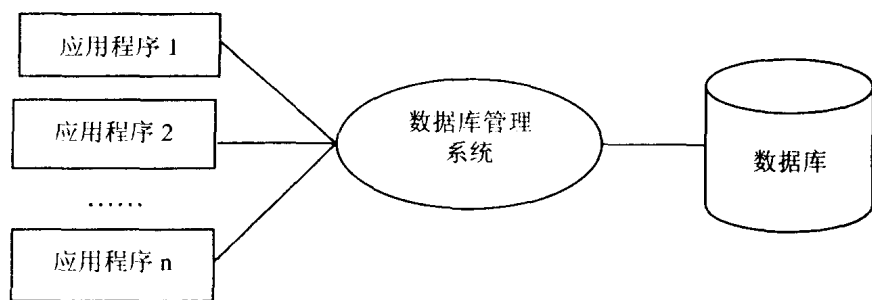


图 1-1 数据库与应用程序的关系

数据库系统是将所有的数据集中到一个数据库中，形成一个数据中心，实行统一规划，集中管理，用户通过数据库管理系统（Data Base Management System，简称为 DBMS）来使用数据库中的数据。在数据库系统阶段，数据与应用程序的关系如图 1-1 所示。

1. 数据库系统的主要特点

(1) 实现了数据的结构化：在数据库中采用了特定的数据模型组织数据。数据库系统把数据存储于有一定结构的数据库文件中，实现了数据的独立和集中管理，克服了人工管理和文件系统管理的缺陷，大大方便了用户的使用，提高了数据管理的效率。

(2) 实现了数据的共享：数据库中的数据能为多个用户服务。



(3) 实现了数据的独立：用户的应用程序与数据的逻辑结构及数据的物理存储方式无关。

(4) 实现了数据的统一控制：数据库系统提供了各种控制功能，保证了数据的并发控制、安全性和完整性。数据库作为多个用户和应用程序的共享资源，允许多个用户同时访问。并发控制可以防止多用户并发访问数据时而产生的数据不一致性。安全性可以防止非法用户存取数据，完整性可以保证数据的正确性和有效性。

在数据库系统阶段，应用程序和数据完全独立，应用程序对数据管理和访问更加灵活，一个数据库可以为多个应用程序共享，使得程序的编制和效率大大提高，减少了数据的冗余度，实现数据资源共享，提高了数据的完整性、一致性以及数据的管理效率。

2. 数据库系统的分类

数据库系统的分类有多种形式，按数据的存放地点，数据库系统分为集中式数据库系统和分布式数据库系统。

(1) 集中式数据库系统

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。所有的用户在存取和访问数据时，都要访问这个数据库。例如，一个银行储蓄系统，如果系统的数据存放在一个集中式数据库中，所有的储户在存款和取款时都要访问这个数据库。这种方式访问方便，但通信量大，速度慢。

(2) 分布式数据库系统

分布式数据库系统是将多个集中式的数据库通过网络联接起来，使各个结点的计算机可以利用网络通信功能访问其他结点上的数据库资源，使各个数据库系统的数据实现高度的共享。分布式数据库系统是在 20 世纪 70 年代后期开始使用的，由于网络技术的发展为数据库提供了良好的运行环境，使数据库系统从集中式发展到分布式，从主机/终端系统发展到客户机/服务器系统结构。在网络环境中，分布式数据库在逻辑上是一个集中式数据库系统，实际上数据是存储在计算机网络的各个结点上。每个结点的用户并不需要了解他所访问的数据究竟在什么地方，就如同在使用集中式数据库一样，因为在网络上的每个结点都有自己的数据库管理系统，都具有独立处理本地事务的能力，而且这些物理上分布的数据库又是共享资源。分布式数据库特别适合地理位置分散的部门和组织机构，如铁路、民航订票系统，银行业务系统等。

在分布式数据库系统中，数据与应用程序的关系如图 1-2 所示。

分布式数据库系统的主要特点：

- 系统具有更高的透明度。
- 可靠性更高、效率更高。
- 局部与集中控制相结合。
- 系统易于扩展。

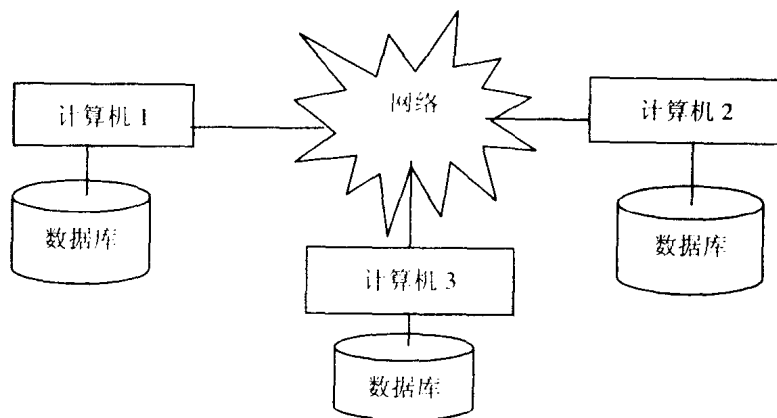


图 1-2 分布式数据库系统数据与应用程序的关系

1.3 数据库系统基本概念

1.3.1 数据库

数据库是按一定的组织形式存储在一起的相互关联的数据集合。实际上，数据库就是一个存放大量业务数据的场所，其中的数据具有特定的组织结构。所谓“组织结构”，是指数据库中的数据不是分散的、孤立的，而是按照某种数据模型组织起来的，不仅数据记录内的数据之间是彼此相关的，数据记录之间在结构上也是有机地联系在一起。数据库具有数据的结构化、独立性、共享性、冗余量小、安全性、完整性和并发控制等基本特点。



1.3.2 数据库管理系统

数据库管理系统是一种负责数据库的定义、建立、操纵、管理、维护的软件系统，是数据库系统的核心部分。数据库管理系统是在操作系统的支持下工作的，它实现了对数据库资源进行统一管理和控制，使数据结构和数据存储具有一定的规范性，提高数据库应用的简明性和方便性。DBMS 为用户管理数据提供了一整套命令，利用这些命令可以实现对数据库的各种操作，如数据结构的定义，数据的输入、输出、编辑、删除、更新、统计、浏览等。

数据库管理系统通常由以下几个部分组成：

- (1) 提供了数据定义语言 DDL (Data Definition Language) 及其编译和解释程序——主要是用于定义数据库的结构。
- (2) 数据操纵语言 DML(Data Manipulation Language)或查询语言——提供了对数据库中的数据存取、检索、统计、修改、删除、输入、输出等基本操作。
- (3) 数据库运行管理和控制例行程序，是数据库管理的核心部分——用于数据的安全性控制、完整性控制、并发控制、通信控制、数据存取、数据库转储、数据库初始装入、数据库恢复、数据库的内部维护等，这些操作都是在该程序控制和统一管理下进行的。
- (4) 数据字典 DD (Data Dictionary) ——提供了对数据库数据描述的集中管理规则，对数据库的使用和操作可以通过查阅数据字典来进行。

1.3.3 数据库系统

数据库系统是指计算机系统引入数据库后的系统构成，是一个具有管理数据库功能的计算机软硬件综合系统。具体地说，它主要包括计算机硬件、操作系统、数据库、数据库管理系统和建立在该数据库之上的相关软件、数据库管理员和用户等组成部分。数据库系统具有数据的结构化、共享性、独立性、可控冗余度以及数据的安全性、完整性和并发控制等特点。

- (1) 硬件系统：是数据库系统的物理支持，包括主机、键盘、显示器、外存储器、输入输出设备等。
- (2) 软件系统：包括系统软件和应用软件。系统软件包括支持数据库管理系统运行的操作系统(如 Windows)和数据库管理系统(如 Visual FoxPro)；应用软件是指在数据库管理系统基础上，用户根据实际问题自行开发的应用程序。
- (3) 数据库是数据库系统的管理对象，为用户提供数据的信息源。
- (4) 数据库管理员是负责管理和控制数据库系统的主要维护管理人员。
- (5) 用户是数据库的使用者，他们利用数据库管理系统软件提供的命令访问数据库并进行各种操作。用户包括专业用户和最终用户。专业用户即程序员，是负责开发应用系统程序的设计人员；最终用户是对数据库进行查询或通过数据库应用系统 (Data Base Application Systems, 简称为 DBAS) 提供的界面使用的数据库人员。

1.3.4 数据库应用系统

数据库应用系统是在 DBMS 支持下根据实际问题开发出来的数据库应用软件，通常是由数据库和应用程序组成。

由于数据库的数据要供不同的应用程序共享，因此在设计应用程序之前首先要对数据库进行设计，数据库的设计是以“关系规范化”理论为指导，按照实际应用的报表数据，首先定义数据的结构，包括逻辑结构和物理结构的定义，然后输入数据形成数据库。开发应用程序也可采用“功能分析，总体设计，模块设计，编码调试”的步骤实现。

1.4 数据模型

数据模型是数据库系统中用于提供信息表示和操作手段的结构形式。简单地说，数据模型是指数据库的组织形式，它决定了数据库中数据之间联系的表达方式。

在数据库系统设计时，首先要确定系统支持的数据模型的类型，从而根据实际情况建立适合某种数据模型的数据库。数据库管理系统所支持的三种主要传统数据模型是：层次模型、网状模型和关系模型。其中层次模型和网状



模型又统称为格式化模型（或非关系模型），早期的数据库系统都是按照格式化模型建立的。

1.4.1 层次模型

用树形结构表示实体及其之间联系的模型称为层次模型。层次模型是数据库系统最早使用的一种模型。层次模型像一棵倒挂的树，根结点在上，是最高层，子结点在下，逐层排列。例如学院中的行政机构、企业中的部门编制等以及数据间的联系都是层次模型。支持层次数据模型的数据库管理系统称为层次数据库管理系统。

1.4.2 网状模型

网状模型是一种比较复杂的数据模型，它是以网状结构表示实体与实体之间的联系。网状模型可以表示多个从属关系的联系，也可以表示数据间的交叉关系，即数据间的横向关系与纵向关系，它是层次模型的扩展。例如铁路运行就是一个网状模型。用于支持网状数据模型的数据库管理系统称为网状数据库管理系统。

1.4.3 关系模型

用二维表的形式表示实体和实体之间联系的数据模型称为关系数据模型。关系模型不像层次模型和网状模型那样使用大量的链接指针把有关数据集合并到一起，而是用一张二维表来描述一个关系。关系模型是将数据组织看成一张二维关系表。用关系模型设计的数据库就是关系型数据库。关系数据库中的每一个关系都是一个二维表。在二维表中，每一行称为一个记录，用于表示一组数据项，表中的每一列称为一个字段或属性，用于表示每列中的数据项。表中的第一行称为**字段名**，用于表示每个字段的名称。

表 1-1 所示是一个学生情况表。

表 1-1 学生情况表

学号	姓名	性别	出生年月	入校总分	三好生	特长	照片
s0201101	王小平	男	10/23/1984	590	F		
s0201102	张 强	男	08/12/1984	568	T		
s0201103	刘 雨	女	01/02/1985	565	F		
s0201104	江 冰	男	07/24/1984	570	F		
s0201105	吴红梅	女	05/12/1984	595	F		
s0201106	杜 海	男	12/12/1983	578	F		
s0201107	金 阳	女	11/07/1983	550	T		
s0201108	张 敏	女	09/30/1984	586	F		
s0201109	杨 然	男	02/15/1985	569	F		
s0201110	郭晨光	男	03/18/1983	592	F		

在关系模型中，二维表应具有以下特征：

- 表中的每一数据项不能再分。
- 表中每一列的数据类型必须相同。
- 表中每一列的字段名不允许相同。
- 表中的记录和字段顺序可以任意排列。

支持关系模型的数据库管理系统称为**关系数据库管理系统**。Visual FoxPro 是一个关系数据库管理系统。



1.5 关系数据库基础知识

1.5.1 关系术语

关系是建立在数学集合概念基础之上，它是由行和列的二维表组成。

关系——指一个二维表，在 Visual FoxPro 中一个关系就称为一个数据表。

元组——指表中一行上的所有数据，在 Visual FoxPro 中一行称为一个记录。

属性——指表中的一列，在 Visual FoxPro 中一列称为一个字段。

域——指表中属性的取值范围。在 Visual FoxPro 中，字段的取值范围称为一个字段的宽度。

关键字——是指表中具有惟一标识的属性，在 Visual FoxPro 中具有惟一性取值的字段称为关键字段。

关系模式——是对关系的描述，一个关系模式对应一个关系的结构。

关系模式的格式：关系名 (属性 1, 属性 2, 属性 3, ..., 属性 n)

例如，学生情况表的关系模式描述如下：

学生情况表 (学号, 姓名, 性别, 出生年月, 入校总分, 三好生, 特长, 照片)

1.5.2 关系运算

在关系数据库中，经常需要对关系进行特定的关系运算操作。基本的关系运算有三种：选择、投影和连接。

1. 选择

选择运算是从关系中找到满足条件的记录。选择运算是一种横向的操作，它可以根据用户的要求从关系中筛选出满足一定条件的记录，这种运算可以改变关系表中的记录个数，但不影响关系的结构。在 Visual FoxPro 的命令中，可以通过短语 FOR<条件>、WHILE<条件>等实现选择运算。

例如，从学生情况表中找出入校总分大于 580 分的学生，如表 1-2 所示。通过 Visual FoxPro 的命令可以从学生情况表中找出入校总分大于 580 分的 4 个记录（灰色记录）。

表 1-2 学生情况表

学号	姓名	性别	出生年月	入校总分	三好生	特长	照片
s0201101	王小平	男	10/23/1984	590	F		
s0201102	张 强	男	08/12/1984	568	T		
s0201103	刘 雨	女	01/02/1985	565	F		
s0201104	江 冰	男	07/24/1984	570	F		
s0201105	吴红梅	女	05/12/1984	595	F		
s0201106	杜 海	男	12/12/1983	578	F		
s0201107	金 阳	女	11/07/1983	550	T		
s0201108	张 敏	女	09/30/1984	586	F		
s0201109	杨 然	男	02/15/1985	569	F		
s0201110	郭晨光	男	03/18/1983	592	F		

2. 投影

投影运算是从关系中选择若干属性（字段）组成新的关系。投影运算是一种纵向操作，即从列的角度进行的运算。经过投影运算可以得到一个新的关系，其关系模式所包含的属性(字段)个数往往比原关系少，或者属性(字段)的排列顺序不同。投影操作可以改变关系的结构。在 Visual FoxPro 的命令中，可以通过短语 FIELDS<字段 1, 字段 2.....>等实现投影运算。

例如，通过 Visual FoxPro 的命令可以从学生情况表 (学号, 姓名, 性别, 出生年月, 入校总分, 三好生, 特长, 照片) 关系中只显示姓名、性别、入校总分三个字段。如表 1-3 所示。



表 1-3 学生情况表

学号	姓名	性别	出生年月	入校总分	三好生	特长	照片
s0201101	王小平	男	10/23/1984	590	F		
s0201102	张 强	男	08/12/1984	568	T		
s0201103	刘 雨	女	01/02/1985	565	F		
s0201104	江 冰	男	07/24/1984	570	F		
s0201105	吴红梅	女	05/12/1984	595	F		
s0201106	杜 海	男	12/12/1983	578	F		
s0201107	金 阳	女	11/07/1983	550	T		
s0201108	张 敏	女	09/30/1984	586	F		
s0201109	杨 然	男	02/15/1985	569	F		
s0201110	郭晨光	男	03/18/1983	592	F		

3. 连接

连接运算是两个关系通过共同的属性名(字段名)进行投影操作来连接生成一个新的关系。这个新的关系可以反映出原来两个关系之间的联系。选择和投影运算均属于一维运算,其操作对象只是一个关系,相当于对一个二维表进行切割。而连接运算是二维运算,需要两个关系作为操作对象,相当于对两个二维表进行拼接。

1.5.3 关系数据库

关系数据库是由若干依照关系模型设计的二维数据表文件的集合。在 Visual FoxPro 中,一个关系数据库是由若干个数据表组成,每个数据表又是由若干个记录组成,每个记录由若干个数据项组成。

关系数据库的每个数据表中的数据如何收集,如何组织,这是一个很重要的问题。当许多相关的数据集合到一个二维表后,数据的关系就会变得很复杂,表中的字段个数及数据量都会使数据出现大量的重复,因此,要求数据库的数据要实现规范化,形成一个组织良好的数据库。

数据的规范化基本思想是逐步消除数据依赖关系中的不合适部分,使得依赖于同一个数据模型的数据达到有效的分离。每一个数据表具有独立的属性,同时又依赖于共同关键字。例如下列五张数据表,收集有关学生选课的情况。

(1) 学生情况表记录了有关学号、姓名、性别、出生年月、入校总分、三好生、特长、照片等情况,如表 1-4 所示。

表 1-4 学生情况表

学号	姓名	性别	出生年月	入校总分	三好生	特长	照片
s0201101	王小平	男	10/23/1984	590	F		
s0201102	张 强	男	08/12/1984	568	T		
s0201103	刘 雨	女	01/02/1985	565	F		
s0201104	江 冰	男	07/24/1984	570	F		
s0201105	吴红梅	女	05/12/1984	595	F		
s0201106	杜 海	男	12/12/1983	578	F		
s0201107	金 阳	女	11/07/1983	550	T		
s0201108	张 敏	女	09/30/1984	586	F		
s0201109	杨 然	男	02/15/1985	569	F		
s0201110	郭晨光	男	03/18/1983	592	F		



(2) 选课表记录了有关学号、课程号、成绩等情况, 如表 1-5 所示。

表 1-5 选课表

学号	课程号	成绩
s0201101	c110	90
s0201101	c120	87
s0201102	c110	80
s0201102	c150	66
s0201103	c130	94
s0201103	c140	50
s0201104	c110	76
s0201105	c120	82
s0201105	c140	70
s0201106	c110	86
s0201106	c150	80
s0201107	c130	90
s0201108	c110	60
s0201108	c140	86
s0201109	c140	76
s0201109	c150	80
s0201109	c160	90
s0201110	c120	78
s0201110	c140	72
s0201110	c160	90

表 1-6 授课表

(3) 授课表记录了有关教师号、课程号等情况, 如表 1-6 所示。

教师号	课程号
t1101	c110
t1102	c150
t1102	c160
t1103	c120
t1103	c140
t1103	c160
t1104	c130
t1105	c120
t1105	c140
t1105	c160



(4) 教师情况表记录了有关教师号、姓名、性别、职称、工资、政府津贴等情况,如表 1-7 所示。

表 1-7 教师情况表

教师号	姓名	性别	职称	工资	政府津贴
t1101	周 密	男	教授	3000.00	T
t1102	陈 静	女	讲师	1500.00	F *
t1103	孙力波	男	副教授	2000.00	F
t1104	肖 君	女	教授	3000.00	T
t1105	赵 辉	男	讲师	1500.00	F

(5) 课程表记录了有关课程号、课程名、课时等情况,如表 1-8 所示。

表 1-8 课程表

课程号	课程名	课时
c110	数学建模	80
c120	计算机网络	60
c130	日语	80
c140	数据库	60
c150	商业会计	70
c160	电子商务	50

上述五张数据表收集了有关学生情况、选课情况、授课情况、教师情况、课程情况等信息。如果将这些数据集中在一个表中,显然会使得表中的数据字段太宽、数据量大、结构复杂,使数据可能重复出现,数据的输入、修改和查找都很麻烦,也会造成数据的存储空间的浪费。

在关系数据库中,通过数据库管理系统,可将这些相关的数据表存储在同一个数据库中,在两个数据表中具有相同值的字段名之间建立关联关系。如将学生情况表中的“学号”字段与选课表中的“学号”字段建立关联关系;将选课表中的“课程号”字段与授课表中的“课程号”字段建立关联关系。使每个数据表具有独立性,又使每个数据表保持一定的关联关系。

1.5.4 关系的完整性

数据库系统在运行的过程中,由于数据输入错误、程序错误、使用者的误操作、非法访问等各方面原因,容易产生数据错误和混乱。为了保证关系中数据的正确、有效使用,需建立数据完整性的约束机制来加以控制。

关系的完整性是指关系中的数据及具有关联关系的数据间必须遵循的制约和依存关系,以保证数据的正确性、有效性和相容性。关系的完整性主要包括实体完整性、域完整性和参照完整性。

1. 实体完整性

实体是关系描述的对象,一行记录是一个实体属性的集合。在关系中用关键字来惟一地标识实体,关键字也就是关系模式中的主属性。实体完整性是指关系中的主属性值不能取空值(NULL)且不能有相同值。若主属性取空值,则不可区分现实世界存在的实体。例如,学生的学号、职工的职工号一定都是惟一的,这些属性都不能取空值。

2. 域完整性

域完整性约束也称为用户自定义完整性。域完整性约束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。

域是关系中属性值的定义范围。域完整性是对数据表中字段属性的约束,它包括字段的值域、字段的类型及字段的有效规则等约束,它是由确定关系结构时所定义的字段的属性所决定。在设计关系模式时,定义属性的类型、宽度是基本的完整性控制。进一步的控制可保证输入数据的合理有效,如性别属性只允许输入“男”或“女”,其他字符的输入则认为无效输入,拒绝接受。Visual FoxPro 命令中的 VALID 语句可进行这方面的控制。