



全国高等农业院校教材
全国高等农业院校教学指导委员会审定



Visual FoxPro 6.0

程序设计基础

盛苏伦 主编

中国农业出版社

内 容 提 要

本书是全国高等农业院校十五规划教材。本书从初学者的角度出发,用通俗易懂的语言、直观详尽的视图,并以一个统一的开发实例“教学管理系统”为主线,系统地介绍了可视化编程技术和面向对象程序设计方法。最后,将全书的内容和思想以一个完整案例为主线进行了综合阐述,起到了画龙点睛的作用。

全书介绍了 VFP 6.0 的程序设计开发环境、VFP 6.0 的基本元素、表与数据库、查询和视图、表单设计、建立报表与标签、菜单与工具栏设计和程序设计等内容。每一章内容都包含了详细的操作步骤和丰富的实例,并配有习题和上机实验题。书后附录含有 VFP 6.0 的常用命令和函数以及常用类的属性、事件和方法。

本书由浅入深,内容丰富新颖,易学易用,可作为高等学校本、专科的教材,也可供从事计算机应用和开发的各类人员学习使用。本书不但适合于 VFP 6.0 的初学者作为入门学习的指导书,对已有一定使用 VFP 6.0 经验的读者来说,也同样有借鉴意义。

另外,本书的编写也兼顾了教育部考试中心制定的《全国计算机等级考试二级大纲(Visual FoxPro 程序设计)》的要求,因此也可以作为计算机等级考试教材之一。

主 编 盛苏伦

副主编 翁 梅 杨林楠

编 者 (按姓氏笔画排序)·

李晗静 杨怀脚 杨林楠

翁 梅 钱 巍 盛苏伦

前 言

数据库技术从早期的 dBASE、FoxBASE 发展到现在的 VFP 6.0，经过了不断的变革，特别是面向对象可视化语言的出现，以其友好的工作界面、可视化的编程技术，为广大数据库开发人员提供了强有力的支持工具。VFP 6.0 中提供了结构化程序设计和面向对象程序设计这两种设计方式，人们正在由旧的结构化设计思想转向全新的面向对象程序设计思想。本书先从数据库基本原理、概念出发，介绍数据表，以及数据库对象的建立、查看、修改、使用与维护等操作，然后在学习结构化程序设计的结构与基本方法的基础上，由浅入深地引入面向对象程序设计思想和可视化的编程技术，更以其简单易学的向导、设计器等开发工具的介绍，使应用程序的开发变得简单、高效、易行。

本书共 9 章内容，第一章介绍 VFP 6.0 基本知识，包括数据与信息、数据库处理、关系型数据库、VFP 6.0 数据库管理系统的功能和特点，以及常量、变量、表达式及函数等内容。第二章介绍 VFP 6.0 用户界面与项目管理器，包括 VFP 6.0 的启动与退出、界面的组成、功能使用和项目管理员的用法等。第三章介绍数据表和数据库的基本操作，包括使用向导、设计器和命令建立表 and 数据库，以及数据库、表的操作、索引、关联、有效性规则、参照完整性规则等内容。第四章介绍查询和视图的建立、使用方法和用途，介绍查询语言 SQL 在 VFP 6.0 中的用法。第五章介绍表单的设计过程、控件的应用、面向对象的基本概念及应用等。第六章介绍报表和标签的创建、设计和输出。第七章介绍结构化程序设计的结构和基本方法。第八章介绍如何设计菜单与工具栏，如何使用菜单设计器创建菜单程序以及如何进行程序编译等内容。第九章着重以一个案例为依托，介绍了面向对象事件驱动程序的设计方法与技巧，以及各种面向对象由事件驱动的简单程序设计等内容。此外，本书的最后还增加了常用命令、常用函数和常用类的属性、事件和方法等 3 个附录。

本书是全国高等农业院校教学指导委员会制订的全国高等农业院校十五规划教材，本书主编盛苏伦，副主编翁梅、杨林楠，参加并完成对本书各章节的制订、修改、审阅和最终定稿等工作。其中第一章由山西农业大学的杨怀卿编

写；第二章、第七章、第八章的第三节和第九章由云南农业大学的杨林楠编写；第三章由东北农业大学的盛苏伦编写；第四章由河南农业大学的翁梅编写；第五章及附录由东北农业大学的钱巍编写；第六章和第八章前两节由哈尔滨工业大学的李哈静编写。各章配有丰富的例题、习题和上机练习。

本书充分考虑到系统性和完整性，不但介绍了使用菜单设计器及表单设计器开发的应用程序的简便方法，还介绍了全程序方式的系统开发方法，使教程内容更加全面和实用。本书的特点是：内容丰富、通俗易懂、实例充足、图文并茂、讲练结合、符合认知规律。书中强调可视化的编程技术和面向对象程序设计方法，并成功地将结构化程序设计和面向对象程序设计有机地结合起来，使初学者可以轻松地学会 VFP 6.0 程序设计。

为适应高等学校学生参加全国计算机等级考试的需求，我们结合教育部考试中心制定的《全国计算机等级考试二级考试大纲》（Visual FoxPro 程序设计），适当调整了本书的内容。

由于编者水平有限，难免有不当之处，敬请读者指正。

编 者

2003 年 10 月

目 录

前 言

第一章 中文 VFP 6.0 基础知识	1
第一节 数据库系统的基本概念	1
第二节 VFP 6.0 的基本功能和特点	10
第三节 VFP 6.0 的数据类型与文件类型	18
第四节 VFP 6.0 的基本元素	20
习题一	41
第二章 用户界面与项目管理器	43
第一节 用户界面	43
第二节 项目管理器	60
习题二	67
上机实验	67
第三章 数据库与数据表	68
第一节 建立表	68
第二节 表的修改和使用	82
第三节 建立数据库	96
第四节 索引、排序和统计	113
第五节 表间关系与参照完整性	122
第六节 数据共享	128
习题三	139
上机实验	141
第四章 查询和视图	142
第一节 基本概念	142

第二节 创建查询	151
第三节 创建视图	178
习题四	192
上机实验	194
第五章 表单设计	195
第一节 基本概念	195
第二节 表单设计	198
第三节 常用的基本控件	226
第四节 MessageBox() 函数	236
习题五	238
上机实验	238
第六章 报表与标签	239
第一节 报表和标签设计技术	239
第二节 建立报表	240
第三节 预览、保存与打印报表	260
第四节 标签的建立和使用	263
习题六	267
上机实验	267
第七章 程序设计	268
第一节 程序设计基础	268
第二节 结构化程序设计应用举例	283
第三节 面向对象的程序设计	286
习题七	292
上机实验	293
第八章 菜单与工具栏	295
第一节 设计菜单	295
第二节 设计工具栏	303
第三节 编译应用程序	310
习题八	319
上机实验	319

第九章 数据库应用程序开发实例	321
第一节 教学管理系统的设计	321
第二节 教学管理系统的实现	324
习题九	339
上机实验	339
附录 A 常用命令	340
附录 B 常用函数	348
附录 C 常用类的属性、事件和方法	353
参考文献	362

第一章 中文 VFP 6.0 基础知识

随着计算机技术的发展,计算机的应用逐渐向非数值计算的各个领域渗透,已经进入社会的各个方面乃至人们的家庭生活中。计算机技术的高速发展推动人类社会步入了一个崭新的信息时代。

数据库技术是数据信息管理技术的重要成果,被广泛地应用于国民经济、文化教育、企业管理以及办公自动化等方面,为计算机的应用开辟了广阔的天地。从某种意义上来说,数据库管理系统是计算机技术和信息时代相结合的产物,是信息处理和数据处理的核心,是研究数据共享的学科,也是计算机科学的一个重要分支。

Visual FoxPro 6.0 (以下简称 VFP 6.0) 是 Microsoft 公司开发的与 Visual C++、Visual J++、Visual Basic 等软件系统捆绑销售的关系型数据库软件系统。本章简要介绍数据库系统的基本概念和中文 VFP 6.0 的基础知识。

第一节 数据库系统的基本概念

一、数据、信息、数据处理

(一) 数据 (Data)

数据是人们为了进行交流、通信、解释,用来表示主客观对象的一种形式。或者说,数据是用来表示数量、活动、事务等情况的一组符号,这些符号可以是文字、字符、数字、表格、图形、声音等。在数据库中,把计算机能识别并能处理的一切符号,都称为数据。通常可将数据分为许多类型,如将表示工资、物价、考试成绩等具有量的多少的数据,称为数值型数据;把表示姓名、商品名称、课程名称等的的数据,称为字符型数据等等。

(二) 信息 (Information)

信息是客观事物属性的反映,是经过加工并对人类社会实践和生产经营活动产生影响的数据表现形式。或者说,信息是对原始数据加工后得到的,是对

于某个目的来说有用的知识或数据。数据是信息的原始资料，数据经过解释并赋予一定的意义后，便成为信息。

(三) 数据处理 (Data process)

数据处理又称为信息处理，即利用计算机对各种类型的数据进行处理，包括对数据的采集、存储、整理、分类、排序、统计、加工、检索、维护和传输等一系列操作过程。数据处理的目的是从大量的、原始的数据中获得我们所需要的资料并从中提取有用的数据成分，作为行为和决策的依据。

二、数据库、数据库管理系统、数据库系统

(一) 数据库 (Database, 简称 DB)

收集并抽取出一个应用所需要的大量数据之后，应将其保存起来以供进一步加工处理和抽取有用信息。数据的保存方法有很多种，如人工保存、存放在文件里、存放在数据库里等，其中数据库是存放数据的最佳场所。

所谓数据库就是长期储存在计算机内、有组织的、可共享的数据的集合。数据库中的数据按一定的数据模型组织、描述和储存，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。

(二) 数据库管理系统 (Database Management System, 简称 DBMS)

收集并抽取出一个应用所需要的大量数据之后，如何科学地组织这些数据并将其存储在数据库中，又如何高效地处理这些数据呢？完成这个任务的是一个软件系统——数据库管理系统。

数据库管理系统是位于用户与操作系统之间的一层数据管理软件。

数据库在建立、运用和维护时由数据库管理系统统一管理、统一控制。数据库管理系统使用户能方便地定义数据和操纵数据，并能够保证数据的安全性、完整性、多用户对数据的并发使用及发生故障后的系统恢复。

(三) 数据库系统 (Database system, 简称 DBS)

数据库系统是指在计算机系统中引入数据库后的系统构成，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，这些人称为数据库管理员 (Database Administrator, 简称 DBA)。

在不致引起混淆的情况下，人们常常把数据库系统简称为数据库。

三、数据库技术的产生与发展

数据库技术是应数据管理任务的需要而产生的。

数据管理是指对数据进行分类、组织、编码、储存、检索和维护，它是数据处理的中心问题。随着计算机硬件和软件的发展，数据管理经历了人工管理、文件系统和数据库系统 3 个发展阶段。

（一）人工管理阶段

在 20 世纪 50 年代中期以前，计算机主要用于科学计算。当时的硬件状况是，外存只有纸带、卡片和磁带，没有磁盘等直接存取的存储设备；软件状况是，没有操作系统，没有管理数据的软件；数据处理方式是批处理。

人工管理数据具有如下特点：

1. 数据不保存 由于当时计算机主要用于科学计算，一般不需要将数据长期保存，只是在计算某一问题时将数据输入，用完就撤走。不仅对用户数据如此处置，对系统软件有时也是这样。

2. 数据需要由应用程序自己管理 由于没有相应的软件系统负责数据的管理工作，应用程序中不仅要规定数据的逻辑结构，而且要设计物理结构，包括存储结构、存取方法、输入方式等，因此程序员负担很重。

3. 数据不共享 数据是面向应用的，一组数据只能对应一个程序。当多个应用程序涉及某些相同的数据时，由于必须各自定义，无法互相利用、互相参照，因此程序与程序之间有大量的冗余数据。

4. 数据不具有独立性 数据的逻辑结构或物理结构发生变化后，必须对应用程序做相应的修改，这就进一步加重了程序员的负担。

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。

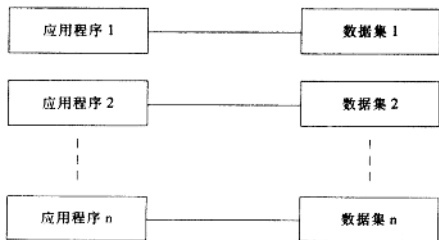


图 1-1 人工管理阶段应用程序与数据之间的对应关系

（二）文件系统阶段

20 世纪 50 年代后期到 60 年代中期，计算机的应用范围逐渐扩大，计算机不仅用于科学计算，而且还大量用于管理。这时硬件上已有了磁盘、磁

鼓等直接存取存储设备；软件方面，操作系统中已经有了专门的数据管理软件，一般称为文件系统；处理方式上不仅有了文件批处理，而且能够联机实时处理。

用文件系统管理数据具有如下特点：

1. 数据可以长期保存 由于计算机大量用于数据处理，数据需要长期保留在外存上，反复进行查询、修改、插入和删除等操作。

2. 由专门的软件即文件系统进行数据管理 程序和数据之间由软件提供的存取方法进行转换，使应用程序与数据之间有了一定的独立性，程序员可以不必过多地考虑物理细节，将精力集中于算法。而且数据在存储上的改变不一定反映在程序上，大大节省了维护程序的工作量。

3. 数据共享性差 在文件系统中，一个文件基本上对应于一个应用程序，即文件仍然是面向应用的。当不同的应用程序具有部分相同的数据时，也必须建立各自的文件，而不能共享相同的数据，因此数据的冗余度大，浪费存储空间。同时由于相同数据的重复存储、各自管理，给数据的修改和维护带来了困难，容易造成数据的不一致性。

4. 数据独立性低 文件系统中的文件是为某一特定应用服务的，文件的逻辑结构对该应用程序来说是优化的，因此要想对现有的数据再增加一些新的应用会很困难，系统不容易扩充。一旦数据的逻辑结构改变，必须修改应用程序，修改文件结构的定义。而应用程序的改变，例如，应用程序改用不同的高级语言等，也将引起文件的数据结构的改变。因此数据与程序之间仍缺乏独立性。可见，文件系统仍然是一个不具有弹性的无结构的数据集合，即文件之间是孤立的，不能反映现实世界事物之间的内在联系。

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。

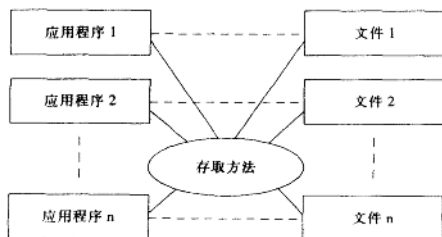


图 1-2 文件系统阶段应用程序与数据之间的对应关系

(三) 数据库系统阶段

20 世纪 60 年代后期以来, 计算机用于管理的规模更为庞大, 应用越来越广泛, 数据量急剧增长, 同时多种应用、多种语言互相覆盖, 共享数据集合的要求越来越强烈。这时硬件已有大容量磁盘, 硬件价格下降, 软件价格上升, 为编制和维护系统软件及应用程序所需的成本相对增加; 在处理方式上, 联机实时处理要求更多, 并开始提出和考虑分布处理。在这种背景下, 以文件系统作为数据管理手段已经不能满足应用的需求, 于是为解决多用户、多应用共享数据的需求, 使数据为尽可能多的应用服务, 就出现了数据库技术, 出现了统一管理数据的专门软件系统——数据库管理系统。

用数据库系统来管理数据具有如下特点:

1. 数据结构化 在文件系统中, 相互独立文件的记录内部是有结构的, 但记录之间没有联系。数据库系统实现了整体数据的结构化, 这是数据库的主要特征之一, 也是数据库系统与文件系统的本质区别。

在数据库系统中, 不仅数据是结构化的, 而且存取数据的方式也很灵活, 可以存取数据库中的某一个数据项、一组数据项、一个记录或一组记录。而在文件系统中, 数据的最小存取单位是记录, 粒度不能细到数据项。

2. 数据的共享性好, 冗余度低 数据的共享程度直接关系到数据的冗余度。数据库系统从整体角度看待和描述数据, 数据不再面向某个应用而是面向整个系统。这样既可以大大减少数据冗余, 节约存储空间, 又能够避免数据之间的不相容性与不一致性。所谓数据的不一致性是指同一数据不同拷贝的值不一样。采用人工管理或文件系统管理时, 由于数据被重复存储, 当不同的应用和修改不同的拷贝时就易造成数据的不一致。

3. 数据独立性高 数据库系统提供了两方面的映象功能, 从而使数据既具有物理独立性, 又有逻辑独立性。

数据库系统的一个映象功能是数据的总体逻辑结构与某类应用所涉及的局部逻辑结构之间的映象或转换功能。这一映象功能保证了当数据的总体逻辑结构改变时, 通过对映象的相应改变可以保持数据的局部逻辑结构不变, 由于应用程序是依据数据的局部逻辑结构编写的, 所以应用程序不必修改。这就是数据与程序的逻辑独立性, 简称数据的逻辑独立性。

数据库系统的另一个映象功能是数据的存储结构与逻辑结构之间的映象或转换功能。这一映象功能保证了当数据的存储结构(或物理结构)改变时, 通过对映象的相应改变可以保持数据的逻辑结构不变, 从而应用程序也不必改变。这就是数据与程序的物理独立性, 简称数据的物理独立性。

数据与程序之间的独立性, 使得可以把数据的定义和描述从应用程序中分

离出去。另外,由于数据的存取由 DBMS 管理,用户不必考虑存取路径等细节,从而简化了应用程序的编制,大大减少了应用程序的维护和修改。

4. 数据由 DBMS 统一管理和控制 由于对数据实行了统一管理,而且所管理的是有结构的数据,因此在使用数据时可以有灵活的方式,可以取整体数据的各种合理子集用于不同的应用系统,而且当应用需求改变或增加时,只要重新选取不同子集或者加上一小部分数据,便可以有更多的用途,满足新的要求。因此使数据库系统弹性大,易于扩充。

除了管理功能以外,为了适应数据共享的环境,DBMS 还必须提供以下几方面的数据控制功能。

(1) 数据的安全性 (security)。数据的安全性是指保护数据,防止不合法使用数据造成数据的泄密和破坏,使每个用户只能按规定对某些数据以某些方式进行访问和处理。

(2) 数据的完整性 (integrity)。数据的完整性是指数据的正确性、有效性和相容性。即将数据控制在有效的范围内,或要求数据之间满足一定的关系。

(3) 并发 (concurrency) 控制。当多个用户的并发进程同时存取、修改数据库时,可能会发生相互干扰从而得到错误的结果,并使得数据库的完整性遭到破坏,因此必须对多用户的并发操作加以控制和协调。

(4) 数据库恢复 (recovery)。计算机系统的硬件故障、软件故障、操作员的失误以及故意的破坏也会影响数据库中数据的正确性,甚至造成数据库部分或全部数据的丢失。这就要求数据库系统必须具有将数据库从错误状态恢复到某一已知的正确状态 (也称为完整状态或一致状态) 的功能,这就是数据库的恢复功能。

数据库管理阶段应用程序与数据之间的对应关系可用图 1-3 表示。

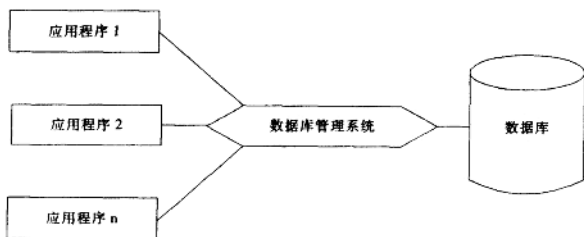


图 1-3 程序与数据的对应关系

综上所述,数据库是长期存储在计算机内有组织的大量的共享数据的集合。它可以供各种用户共享,具有最小冗余度和较高的数据独立性。数据库系统在数据库建立、运用和维护时对数据库进行统一控制,以保证数据的完整性、安全性,并在多用户同时使用数据库时进行并发控制,在发生故障后对系统进行恢复。

20 世纪 60 年代末出现了第一代数据库——网状数据库、层次数据库,20 世纪 70 年代出现了第二代数据库——关系数据库。目前关系数据库系统已逐渐淘汰了网状数据库和层次数据库,成为当今最为流行的商用数据库系统。而 20 世纪 80 年代出现的以面向对象模型为主要特征的数据库系统又在向关系数据库系统挑战。数据库技术与网络通信技术、人工智能技术、面向对象程序设计技术、并行计算技术等互相渗透、互相结合,成为当前数据库技术发展的主要特征。

四、数据模型

数据库系统研究的对象是客观事物以及反映这些客观事物间相互联系的数据。数据库把数据按一定结构和形式组织起来,各个数据对象以及它们之间存在的相互关系的集合,称为数据模型。

数据模型反映了客观世界中各种事物的联系,是用来反映数据与数据之间联系的静态结构与行为的一种手段。由于数据库中的数据都是结构化的数据,因此,讨论数据模型对数据库设计来说很重要。

根据数据所描述实体对象的属性特征以及人们所采用的描述方法,数据模型可分为 3 种:层次模型、网状模型和关系模型。相应地,数据库也可分为层次型数据库、网状型数据库和关系型数据库。

(一) 层次模型

在层次模型中,每个数据元素可以与下面任何一层的多个数据元素相联系,但只能与它上面一层中的一个数据元素相联系。层次模型最简单的结构及约束描述,可以用数据结构中的树来描述。层次模型像一棵由根元素出发向下逐层辐射枝叶的倒置的树,最高一层的数据元素称为根元素,根元素是惟一的,而且只有下属元素。树的结点表示数据元素,枝表示数据元素之间的联系。在层次模型中,有且仅有一个结点无上层结点(根);其他结点有且仅有一个上层结点。如一个学校的组织机构就是层次模型(图 1-4)。

层次模型的特点是层次分明,结构清晰,适于表达事物之间“一对多”型的联系。如一个单位的行政组织关系、生产组织关系等。但是,由于层次模型结构的严格性,它不能直接地表示所有的事实,并且对数据存储和操作都不理

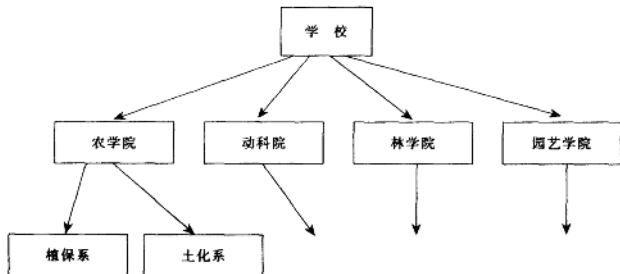


图 1-4 层次模型示意图

想。

(二) 网状模型

在网状模型中，可以有一个以上的数据元素没有上层联系；至少有一个以上结点多于一个的上层结点。因此，网状模型有更一般的表示事实的能力，可以用来描述事物之间的较复杂的联系。图 1-5 所示是网状模型的一个例子，表示在学校模型中的院部、系、教师和学生之间的网状联系。

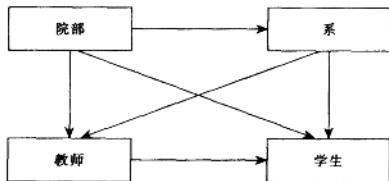


图 1-5 网状模型示意图

网状模型适于表达事物之间的“多对多”型的联系。

(三) 关系模型

在关系模型中，不同数据之间的联系用关系来表示，其实质是将数据的逻辑结构归结为满足一定条件的二维表形式，每个二维表称为一个关系。在二维表中，每一行称为一个元组或记录，每一列称为一个数据项或字段，数据项名称称为字段名或属性名，整个表表示一个关系。这种模型既可用于表达事物之间“一对一”和“一对多”的联系，也可以用来表达“多对多”的联系。

以表 1-1 所示的学生档案关系为例来分析关系数据模型。

关系模型的主要特点有：

(1) 关系中每一列具有相同的属性，每个属性（列）被指定一个不同的属性名（字段名），且属性名不能重复，列数根据需要设置。

(2) 关系中的每一个数据项必须是简单的数据项，而不是组合数据项。

(3) 关系中每一行记录由一个个体事物的诸多属性构成。

(4) 行和列的排列顺序是任意的。

利用关系模型构筑的关系型数据库，可以提供数据库系统的全部功能。

表 1-1 某院学生档案表

学号	姓名	性别	专业	出生日期	入学成绩	党员否
9901001	孙晓慧	女	会计	03/17/82	567.00	F
9804002	刘颖	女	金融	12/07/80	580.50	T
0003021	张军	男	工商管理	08/15/83	600.50	T
9801015	郑学义	男	会计	06/11/81	530.40	F
0105002	沈碧荣	女	经济管理	02/12/83	540.00	F
0102010	连成龙	男	信息管理	12/23/84	660.00	T

五、关系数据库

（一）关系数据库的结构

关系数据库 (Relation database) 是若干个按照关系模型设计的数据表文件的集合，也可以说，关系数据库是由若干张按关系模型设计的二维表组成的。

一般将一张二维表称为一个数据表，在数据表中包含了数据和数据之间的关系。一个关系数据库由若干个数据表组成，数据表由若干个记录构成，而每一条记录又由若干个字段（数据项）组成。在关系数据库中，每一个数据表都具有相对独立性，该独立性的惟一标志是数据表名，又称为表文件名。一个表文件名代表一个独立的表文件。在关系数据库中，对数据表中数据的访问首先通过表文件名实现，因而不允许有同名数据表。关系数据库中各个数据表的独立性，可以方便对数据表中的数据进行操作、存取和传输。

对于关系数据库中具有相关性的数据表，其相关性依靠各个独立数据表内具有相同属性的字段而建立。一般情况下，一个关系数据库中可以有多个相关的独立数据表，可为数据资源的共享和充分利用提供极大的方便。

(二) 关系运算

关系之间可以进行运算，关系运算可以用两种方式表示：关系代数和关系演算。已经证明，这两种方式是等价的。

常用的关系运算有两类：传统的集合运算和专门的关系运算。

1. 传统的集合运算 如集合的并、交、差、广义笛卡尔积等。这类运算将关系看做元组的集合，其运算是从关系的“水平”方向，即二维表的行的角度来进行的。

2. 专门的关系运算 如选择、投影、连接等。这类运算不仅涉及关系的水平方向（二维表的行），而且涉及关系的竖直方向（二维表的列）。

(1) 选择运算 (Selection)。选择运算是在关系中选择满足某些条件的元组，即选择满足条件的行。例如，在学生档案表中选取所有女学生的元组，可用选择运算实现，选择的条件是性别等于“女”。选择运算常用于对指定关系按条件进行检索。

(2) 投影运算 (Projection)。投影运算是在关系中选择某些属性的列，其余属性列都不在运算结果中。投影运算是用来检索指定关系中用户所关心属性列的值，运算结果是一个新的关系。

(3) 连接运算 (Join)。连接运算是在两个关系的笛卡尔积中选取属性间满足一定条件的元组。连接条件中的属性称为连接属性，两个关系中的连接属性应有相同的数据类型，以保证其是可比的。

一般来说，关系代数的运算对象是关系，操作的结果仍然是关系。利用关系代数的运算，可以产生所需要的新关系，或进行指定条件的检索（选择、投影）。对关系数据库的实际操作，常常是以上几种运算的综合应用。

第二节 VFP 6.0 的基本功能和特点

一、VFP 6.0 的基本功能

VFP 6.0 为我们提供了一套用于组织数据、建立查询、生成表单、打印报表、共享数据以及支持超级链接的工具，使用它可以做很多事情。

(一) 组织数据

数据库管理系统最主要的作用就是组织、管理各种各样的数据。VFP 6.0 中的表对象是用于组织数据的基本模块，将每一种类型的数据放在一个表里，如图 1-6 所示。可以定义各个表之间的关系，从而可以将各个表中相关的数据有机地联系在一起。