

# 数据库语言程序设计

主 编 黄维金

副主编 刘永祚 马丁

 **北京理工大学出版社**

BEIJING INSTITUTE OF TECHNOLOGY PRESS

# 内 容 简 介

本书以 Visual FoxPro 6.0 中文版为语言背景，通过实例深入浅出地介绍了 Visual FoxPro 6.0 中数据库的基本概念和基本知识，项目管理器、数据库和表的基本操作和使用，编程的工具与步骤，关系数据库标准语言 SQL 程序设计，面向过程的结构化程序设计及面向对象的可视化程序设计。本书概念清楚、结构严谨、例题丰富、方便实用。其显著特点是它不限于采用可视性操作方式编程，也兼顾面向过程的结构化程序设计。本书内容涵盖了全国计算机等级考试委员会 2001 年修订的二级考试（Visual FoxPro 程序设计）大纲要求的全部内容。读后可全面掌握 Visual FoxPro 6.0 面向过程和面向对象的程序设计方法和技巧。

书中各章均配有习题，书后附有上机实验指导。适合作为大学本科、专科、高职及各类学校的教材，也可作为参加全国计算机等级考试二级（Visual FoxPro 程序设计）的参考教材和自学参考书。

# 前 言

数据库技术是计算机最重要的应用领域之一。它利用计算机存储和管理各种数据，是实现社会信息化的基础。微型计算机上使用的关系数据库管理系统其发展经历了 dBASE、FoxBASE、FoxPro 和 Visual FoxPro 等阶段。程序设计方法，已由面向过程的结构化程序设计方法，发展到面向对象由事件驱动的程序设计方法，这是程序设计思想和程序设计方法的突破性进展。Visual FoxPro 是目前微机上最流行的关系数据库语言之一，它以其面向对象(Object Oriented)和可视化(Visual)的编程技术，卓越的数据处理性能、良好的开发环境、性能完善的编程语言，使其成为在 Windows 平台上广泛使用的数据库开发工具。可视化编程技术为不具备应用程序设计技术的广大用户获得了开发应用程序的能力，也促使了 Visual FoxPro 的推广应用。

本书以 Visual FoxPro 6.0 中文版为支持平台，以程序设计为主线，通过实例讲解了数据库系统与 Visual FoxPro 的基本概念和基础知识，关系型数据库语言程序设计的基础知识、设计思想和方法。计算机基础教育中程序设计课的教学目的主要是对学生进行程序式思维训练和应用知识能力的培养，结构化程序设计语言相对于可视化程序设计语言虽然程序开发效率低，但对初学者学习数据库原理，培养逻辑思维能力，提高自学其他计算机高级语言能力大为有益。Visual FoxPro 在支持面向对象的编程方式的同时，仍然保留了面向过程的设计能力。本书以“数据库语言程序设计”为主题，较系统地介绍了程序设计的概念、思想与方法，程序中常用的算法。程序的基本结构，包括结构化程序设计中数据的输入、输出，顺序结构程序设计，选择结构程序设计，循环结构程序设计，子程序的使用，关系数据库标准语言 SQL 程序设计等。

面向对象的程序设计力图使程序设计环境适合现实世界，允许用户对“对象”(object)和“类”(class)进行定义，并编写相应代码。“可视化编程”作为程序设计的新概念、新方法，全书将面向对象由事件驱动的程序设计方法作为程序设计的重点，把类、对象、事件、属性与方法、视图，以及事件驱动程序等深奥的概念；面向对象程序设计所需的基础知识，常用的 Visual FoxPro 6.0 命令、函数；项目管理器、数据库和表的基本操作与使用；面向对象的编程思想，可视化程序设计的通用方法和步骤等，采用通俗易懂的语言，通过应用实例介绍，使读者易于理解和掌握。

作为计算机基础教材，全书精选内容，深入浅出，通过大量的实例讲解程序设计。既考虑教授内容的基础性，又考虑其实用性、系统性和先进性。在讲授或自学时不需要 FoxBASE 等前导课的知识。

本书内容涵盖了全国计算机等级考试委员会 2001 年修订的二级考试(Visual FoxPro 程序设计)大纲要求的全部内容。各章均配有习题，书后附有上机实验指导，有助于读者理解基本概念，学会操作方法，取得最佳的学习效果。

本书既可作为各类院校及计算机应用人员的培训教材，又可作为参加全国计算机等级考试二级（Visual FoxPro 程序设计）应试者的教学参考书。

全书写作分工如下（按撰写章节先后为序）：黄维金 第一章、第二章，刘永祚 第三章，郭艳玲 第四章，陈长海 第五章，刘永祚 第六章，马丁 第七章，杜彦辉 第八章。上机实验指导由刘永祚和陈长海编写。本书由黄维金统改定稿。

本书在编写过程中，得到了公安大学闫俊杰副校长、李文燕副校长及章晓麟副研究员的大力支持，对此表示衷心的感谢。

全书涉及技术内容广泛，由于作者水平有限，且成书仓促，疏漏之处在所难免，敬请读者不吝赐教。

编 者

2002年7月

# 第一章 数据库基础知识

## 1.1 计算机数据处理技术的发展

计算机是人类社会 20 世纪的重大科技成果之一。它的出现和发展，标志着用机器来存储和处理数据时代的到来，其数据处理的方式也不断改进。

数据处理，也称信息处理或信息加工，是计算机应用的一个重要方面，包括对数据进行收集、存储、传送、组织、检索、计算、输出和维护等。这是近代计算机的应用成果，广泛应用于社会生活的各个领域，各部门管理工作应用计算机就属于这种类型。例如，企业管理、库存管理、人事档案管理、工资管理、报表统计、信息情报检索等。它是计算机最大的应用领域。

随着计算机的发展，数据处理也经历了由低级向高级的发展过程。大体可分三个阶段：

### 1.1.1 人工(程序)管理阶段

20 世纪 50 年代末以前，计算机主要用于科学计算，少量用于数据处理。这时，对于数据保存的需求并不迫切，一般是在算题时，将原始数据与程序一起输入主存，运算处理后将结果数据输出。数据由人工保管，这便是人工管理阶段。其特点是由应用程序来执行数据访问操作，数据结构在应用程序中说明，若改动数据或存储载体，其应用程序也要作相应的改动；缺乏访问控制机制，数据大量冗余；数据的物理存储结构与逻辑结构格式均由编程人员各自设计，并嵌入在程序中。即一个应用程序自带一组数据，程序和数据相互依赖。见图 1.1。

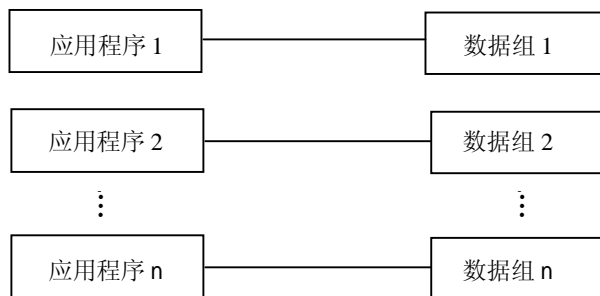


图 1.1 手工管理阶段数据与程序间的关系

### 1.1.2 文件管理阶段

20 世纪 50 年代后期到 60 年代后期，随着计算机应用的普及，数据处理逐渐上升为主要应用领域，该领域需要对大量数据进行存储、检索和维护。因此，数据结构设计与数据管理

技术研究得到迅速发展，出现了文件管理系统。它把数据组织在一个个独立的数据文件中，实现了“按文件名来访问，按记录进行存取”的管理技术，大大减轻了程序员的数据管理劳动。这一阶段技术上有不少改进，文件既可顺序存取，也可随机存取；数据的物理存储结构与逻辑结构不完全一致，修改数据存储单元后，应用程序可不必修改。文件管理阶段与人工管理阶段相比，由于是通过文件系统来管理和使用各种设备介质上的信息，把信息的逻辑结构映像成设备介质上的物理结构。这样就可以使用户不必过多地考虑物理细节，而将精力集中于算法。

文件系统中的数据文件基本上是对应于一个或几个应用程序，实现了以文件形式的数据共享，或者说数据是面向应用程序的，但它仍然是一个不具弹性的无结构的信息集合。至今，文件管理仍是高级语言普遍采用的数据管理方式。见图 1.2。

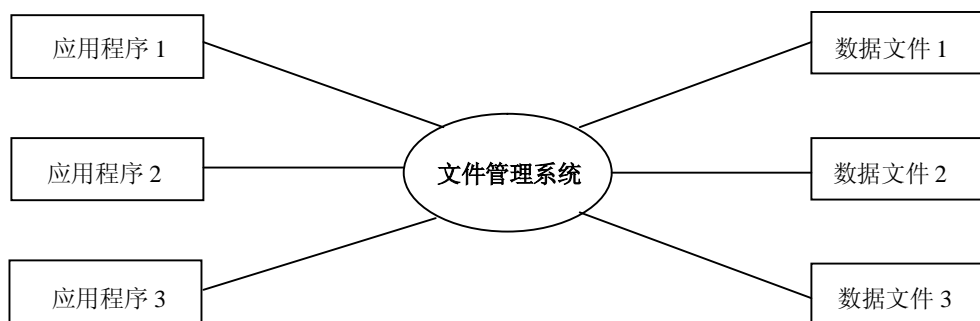


图 1.2 文件系统阶段数据与程序间的关系

该数据处理方式存在以下问题：

#### 一、冗余度大

文件系统下的用户各自建立自己的文件，相互之间数据不能共享，造成数据大量重复存储，不仅浪费存储空间，更严重的是造成数据的不一致性。

#### 二、数据和应用程序独立性差（数据依赖）

由于应用程序和数据结构相互依赖，一旦因某种原因导致数据存储结构改变，必须修改应用程序，修改文件结构的定义。编写程序仍依赖于具体数据，使得应用程序的编制和维护成本大大增加。

#### 三、数据无集中管理

各文件中的数据无集中管理机构，缺乏对数据操作的控制方法，其安全性和完整性无法得到保证。

#### 四、不灵活

每个数据文件仅限于一定的应用，文件不易修改和扩充，不能反映现实世界事物间的内在联系。

### 1.1.3 数据库管理阶段

由于传统文件系统的上述缺点，促使人们寻求新的数据管理技术。20 世纪 70 年代以后，大容量和快速存取磁盘设备的重大进展，为数据管理技术提供了良好的物质基础，数据管理

技术进入了数据库技术时代，出现了数据库系统。数据库系统的出现是对计算机数据处理技术的重要发展，其目标首先就是克服文件系统中的这些弊病，试图提供一种完善的数据管理方式来集中管理所有的数据，以实现数据的高度独立性，实现对数据的统一集中管理和多用户数据共享问题，并保证数据的完整性和安全性。

数据库系统的主要特点：

### 一、对数据实行统一的控制

数据库中的数据是各用户的共享资源，其共享一般是并发的。因此，确保数据的安全和完整十分重要。数据库系统提供统一的数据控制功能，并加入了安全保密机制，允许多个用户同时存取数据而不互相影响，有效地防止了对数据的非法存取。由于对数据进行集中控制和并发访问控制，保证了数据的完整性、安全性和正确性。

### 二、数据独立性

应用程序不必随数据存储结构的变动而变动，这是数据库技术最根本的特征。数据独立性包括：

1. 物理数据独立。数据存储格式和组织方式改变时，不影响数据库的逻辑结构，从而避免影响应用程序。

2. 逻辑数据独立。即数据库逻辑结构的变化（如数据定义的修改、数据间联系的变更等），不会影响用户的应用程序。

### 三、可控冗余度

用户的逻辑数据文件和具体的物理数据文件不必一一对应，存在着“多对一”的重叠关系，有效地节省了存储资源。

### 四、数据结构化

在文件系统中，文件之间不存在联系。文件内部的数据虽有结构，但从数据整体来说是没有结构的。数据库系统在结构上服从一定的结构形式，从而能适应大量数据管理和维护的客观需要。

### 五、避免了数据的不一致性

由于数据只有一个物理备份，数据的访问不会出现不一致的情况。

从文件系统发展到数据库系统是信息处理领域的一个重大变革。数据库系统的核心是数据库管理系统 DBMS（Data Base Management System）。在 DBMS 的集中管理下，将具有较高的数据独立性及较低的冗余度。数据库管理阶段数据与程序间的关系见图 1.3。

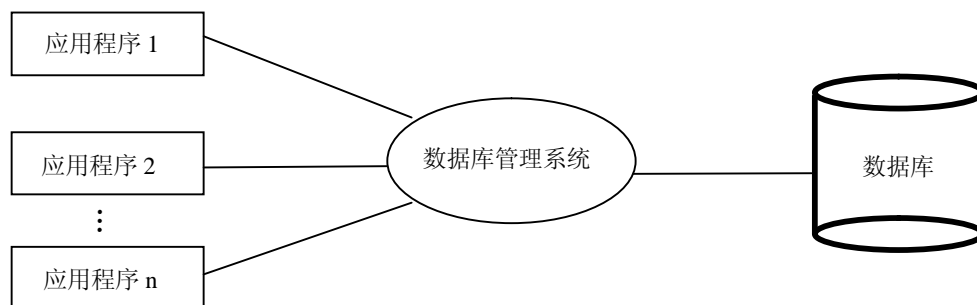


图 1.3 数据库管理阶段数据与程序间的关系

## 1.2 数据库系统的基本概念

在信息化时代，计算机更广泛地应用于信息处理领域。其特点是：数据量大，实时性强，数据必须准确可靠。多采用数据库技术来实现。

数据库是计算机软件的一个重要分支，它和计算机网络、人工智能并称为当今计算机技术的三大热门技术。

数据 (Data)、数据库 (Database)，数据库管理系统 (Data Base Management System)，数据库系统 (Database System) 等是数据库技术中常用的术语。相互之间有一定的区别和联系。

### 一、数据

数据 (data) 是人们对于客观事物进行抽象提取的结果，用于反映客观事物的属性。一般指能够输入到计算机由计算机处理的数字、字母和符号等。数据是数据库中存储的信息，是数据库系统的操作对象。数据不仅包括数值数据，而且包括各种非数值数据。例如，人事档案管理系统中，职工的年龄、工资等是数值型数据，而姓名、出生日期、个人简历等则是非数值数据。使用现代技术，不仅文字可以数字化，而且图形、图像、语音等，也都可以数字化，转换成可以用计算机来处理的数据。

在数据处理中，从数据构成来看，可以将数据分为字符 (character)、字段 (field)、记录 (record)、表 (table) 和数据库 (database) 等由简单到复杂的五个层次，其中，每个层次都是后一层次的组成部分；即后一层次的数据都是前一层次数据的集合。

字符通常是指英文字母、数字和符号，每个字符在计算机中占用 1 个字节 (byte) 存储空间；而在汉字系统中，一个汉字占用 2 个字节，在“全角”状态下的英文字母、数字和符号同汉字一样，也占用 2 个字节。由若干字符可以组成一个字段，例如，可以组成一个职工的编号、姓名、性别、出生日期、个人简历等字段。字段是最基本的、不可分的、有名的数据单位。记录由相关字段组成，例如，由每位职工的编号、姓名、性别、出生日期、婚否、职称、部门、工资、个人简历等字段组成描述一个职工基本情况的记录。若干记录可以组成一个表，它可以以文件 (file) 的形式存储在计算机中。例如，由单位所有职工的记录可以组成该单位的人事档案表，它可能是人事档案管理数据库中的一个表。而在人事档案管理数据库中，可能还包括一些别的表，如工资表、退休职工表、工会会员表等，人事档案管理数据库主要就是由这些表组成。一个数据库可以包含一个或多个数据文件 (表) 及其他一些相关文件。

### 二、数据库

数据库是存储在计算机内有结构数据的集合。形象地说就是存储数据的“仓库”。这里的数据库并不是一些数据的简单堆积，而是指在计算机内存储的相互关联的、有结构的数据组成。存储在数据库中的数据具有集中性和共享性两个特点。其集中性是指可把数据库看成性质不同的数据文件的集成体。所谓共享是指多个不同的用户，使用各种不同的语言，为了不同的应用目的可同时存取数据库中的数据，其中的数据具有极小冗余度。与某种应用有关的数据按一定方式组织起来，具有一定结构和规范，构成表，以文件形式存储在计算机中，再

由这些表和其他相关文件构成数据库。数据库是表和相关文件的集合。

### 三、数据库管理系统

是指帮助用户建立、使用和管理数据库的软件系统，简称 DBMS。其职能是维护数据库，接受和完成用户程序或命令提出的访问数据的各种请求，并负责保护数据的安全性和可靠性，是数据库系统的核心。这个系统是根据数据之间的自然联系结构而成的，从而能提供一切必须的路径存取其中的每一个或任意一部分数据单元，以满足所有用户的要求。用户使用数据库是目的，而数据库管理系统是达到这一目的的工具和手段。

### 四、数据库系统

数据库系统是实现有组织地、动态地存储大量关联数据，方便多用户访问的计算机软件（系统软件和 DBMS）、硬件资源及有关人员（包括系统开发、管理和使用人员）组成的人机系统，即指计算机系统引进数据库后的系统构成。有时也简称“数据库”。如原中国科学院的“科学数据库”。它与文件系统的重要区别是数据的充分共享、交叉访问、与应用（程序）的高度独立性；而且包括设备和人，打个通俗的比方，数据库系统可比作图书馆。图书馆是存储并管理图书的部门，而数据库系统则是存储和管理数据的机构。正如图书馆不能等同于书库一样，也不能把数据库系统仅理解为数据的集合。

### 五、数据库应用系统

数据库应用系统是在 DBMS 支持下运行的一类计算机应用系统，简称 DBAS。通常由数据库、应用程序和支持它们的专用或通用 DBMS 构成。在微机上运行的 DBAS 一般都使用通用的 DBMS（例如 Visual FoxPro DBMS）开发的数据库和应用程序。

### 六、数据库软件

负责对数据库描述、管理和维护的软件。具有对数据的定义、描述、操作、维护、管理和控制等功能。数据库软件的核心是数据库管理系统。

### 七、数据库管理员（DBA—DataBase Administrator）

专门负责数据库的设计、建立、执行和维护的工作人员。

## 1.3 数据库模型

数据库中的数据从整体来看是有结构的，即所谓数据的结构化。这种结构用数据模型来表示，按照结构化所采取的不同联系方式，数据的整体结构可区分为三种基本的数据模型：层次模型、网络模型和关系模型。其中前两类又称为“格式化模型”。构造数据模型，是应用数据库的关键问题。

### 1.3.1 层次模型

层次模型是将数据元素分为若干层，最高一层只有一个数据元素，称为根，每一个数据元素都和下一层的一个或多个元素连接；除了根元素外，其他元素都和上一层一个元素相连接。所有元素按这种方式组成以根元素为起点的树形结构。见图 1.4。

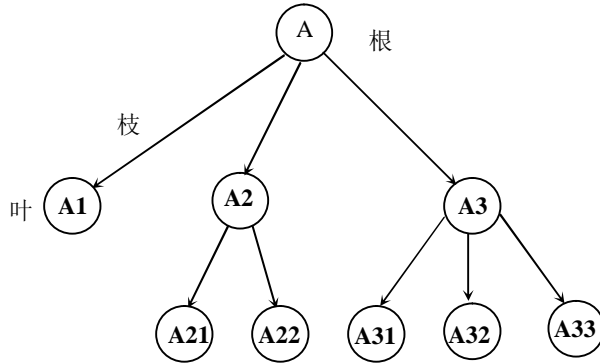


图 1.4 层次式(树型)数据库结构

数据库理论中，一般称上一元素为“双亲”结点，下一元素称为“子女”结点。在该结构中，其基本联系是一对多的联系。层次模型中，有，且仅有一个结点无双亲（树根）；其他结点有且仅有一个双亲。即每一双亲可以有多个子女；但每个子女却只能属于唯一的双亲。

在层次模型中，记录之间只能有单线联系。例如，一个单位的部门建制，就是一种层次模型。以一所大学为例，一所大学下分系→教研室→教员，处→科→科员。层次模型中数据的存取是采用层次路径来表示的。

层次路径表示： $A \rightarrow A1$ ， $A \rightarrow A2 \rightarrow A21$ ， $A \rightarrow A2 \rightarrow A22$  ……

### 1.3.2 网络模型

网络模型中，每一个元素都和任意一个或多个其他数据相连接，形成网络。见图 1.5。

网络模型的特征：

- (1) 可以有一个以上的结点无双亲；
- (2) 至少有一个结点有多于一个的双亲。

即在层次式(树型)数据库中，如果一个子女可以属于几个双亲，其结构就变成网状的了。层次式(树型)数据库和网络式数据库的主要区别：

层次式(树型)数据库从子女到双亲的联系是唯一的；而网络式数据库中从子女到双亲的联系不惟一。

网络模型允许记录之间存在两种或多种的联系，其结构错综复杂，数据存取路径需用户指定，使用难度较高。例如，在联网的火车票销售系统中，如果将每个车站的车票数据作为一个数据元素，则所有火车票销售点的数据就构成一个网络数据模型。网络模型中数据的存取是采用网络路径来表示的。

网络路径表示： $A1 \rightarrow B2 \rightarrow C1$ ， $A2 \rightarrow B2 \rightarrow C1$ ， $A2 \rightarrow B3 \rightarrow C1$  ……

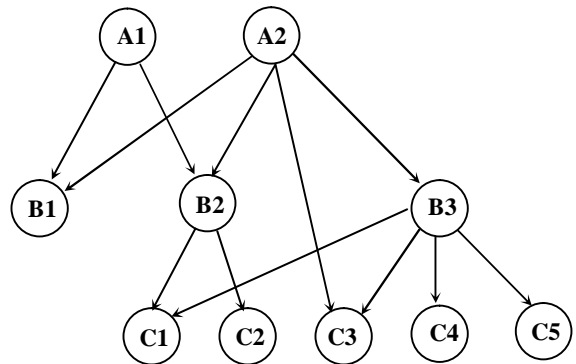


图 1.5 网络式数据库结构

### 1.3.3 关系模型

关系模型是用二维表（表格）来描述相关数据，即用表格来表示数据与数据之间的联系。例如，图 1.6 的人事档案表与职工工资表。关系模型是三种数据模型中最主要的模型。目前微机上实现的 DBMS 软件几乎都是关系型数据库，数据库领域当前的研究工作也多以关系方法为基础。关系型数据库在数据管理领域占主要地位。

所谓关系，实际上就是一个命名的二维表。一般来讲，一个关系型数据库可以看成为一张或几张二维表格。关系式模型是用二维表（表格）来描述相关数据。例如人事档案表、工资表等。

每个二维表称为一个“关系”，它一般存放在数据库文件中，也可以单独存在。在关系模型中，“表”和“关系”实际上指的是同一对象，在使用中往往不加以区分。

客观存在并且可以相互区分的事物称为实体。实体可以是实际事物（例如，一位教师、一支粉笔），也可以是抽象事件（如教师上课，用粉笔书写等）。二维表中存放了实体本身的数据和实体间的联系这两类数据。同类型实体的集合构成一个实体集。

二维表中的每一列称为“属性”，又称为“字段”。属性反映了实体的特性。例如，教师实体可以用编号、姓名、性别、出生日期、职称等属性来描述；粉笔实体可以用颜色、长短、生产厂等属性来描述。表中第一行列出的是属性名（字段名），如编号、姓名、职称等。以下各行中，每一列的内容是属性值（字段值），如图 1.6 中，“0004”、“张明”、“教授”，分别是字段编号、姓名、职称的一个值，它是不可拆分的基本数据项。表中的第二行开始的每一行称为“元组”，又称为“记录”，是构成“关系”的一个个实体。实体（表）之间的对应关系称为联系，它反映现实世界事物之间的相互关联。联系分为两种：实体内部各属性之间的联系和实体与实体之间的联系。实体间的联系是通过不同关系中具有相同属性名来实现的。实体之间的联系有三种类型：一对一联系，一对多联系和多对多联系。在关系模型中，只有单一的“关系”这种结构类型。结构单一化，适合计算机处理。关系模型有着坚实的数学理论基础，关系是数学中集合论的一个概念，对关系可以进行各种运算，运算结果形成新的关系。在关系数据库系统中，对数据的操作多数都可以归纳为对关系的运算。

## 1.4 关系数据库系统

### 1.4.1 关系的数学定义

关系数据模型不同于层次模型和网络模型，它是通过二维表结构来表示实体与实体之间联系的模型，而不是通过指针（路径）连接来表示和实现两个实体间的联系。关系模型迅速得到广泛应用的重要原因，在于它有坚实的数学理论基础。关系模型的数学基础是集合代数理论。

表是一类数据的集合，这类数据应有共同的特征。表格中的数据可看成元素。每一行是一个元组，它相当于一个记录值；每一列是一个属性值集，属性的取值范围叫“域”，属性相当于数据项。元组的集合就是一张二维表，即关系。如果表格有  $n$  列，就称该关系为  $n$  元关

系。关系是元组的集合，元组是属性值的集合。

二维表中存放的数据在数据库系统中以文件方式来存放。表中第二行开始的每一行（元组），在数据库系统中称为一个记录，是构成关系的一个实体。

每一列称为“字段”。每一字段有一名字，称为“字段名”（各表表头），一个表中通常有多个字段。图 1.6 分别为人事档案表和工资表中部分内容。表中的编号、姓名、出生日期、工资等均为字段名，一般简称为“字段”。

编号	姓名	性别	出生日期	婚否	职称	部门	工资
	⋮						
0003	赵龙	男	1963/09/01	T	副教授	基础部	900.00
0004	张明	女	1959/02/16	T	教授	基础部	1200.00
0005	钱民生	男	1966/10/28	T	副教授	基础部	960.00
0006	欧阳一民	男	1976/05/27	F	助教	基础部	600.00
	⋮						

(表一)

编 号	姓 名	基本工资	附加工资	实发工资
	⋮			
0003	赵龙	900.00	250.00	1150.00
0004	张明	1200.00	500.00	1700.00
0005	钱民生	960.00	300.00	1260.00
0006	欧阳一民	600.00	150.00	750.00
	⋮			

(表二)

图 1.6 人事档案与职工工资表

关系模型把数据之间的复杂联系代换成表格之间的关系，但表格之间应有保持联系的信息。如上面两个表之间通过编号保持联系。

每张表格应有如下性质：

1. 每一列的数据应属于同一类型数据。
2. 每张表中的字段名不能重名。
3. 表中不允许有重复行。
4. 每一列都是不能再分的基本信息单位。
5. 表中行和列的位置改变不影响其信息内容。

### 1.4.2 专门的关系操作

关系数据库是建立在数学概念基础上的应用数学理论处理数据库组织的方法。Visual

FoxPro 是一种关系型数据库管理系统，它能完成三种基本关系操作：

(1) 选择 (SELECT)：只对表文件中某些指定的记录进行操作。包括范围筛选和指定条件筛选。选择是从行的角度进行运算，运算条件由逻辑运算符(与、或、非)连接各运算表达式组成。

数学上为：从  $n$  维空间内的所有点中筛选出满足一定条件的那些点。即从给定的关系中选取满足给定条件的那些元组。

(2) 投影(PROJECT)：只对一个记录类型中的某些项进行数据操作。

数学上为：从给定的关系中抽取某些属性组成一个新关系，并去掉其中重复元组的运算。

(3) 连接(JOIN)：指将两个表文件按一定条件联结成一个新的表文件。

数学上为：将两个关系按一定条件连结成一个新的关系。

## 习 题 一

### 一、单项选择题

- 1.数据库管理系统常用的数据模型有 ( )。  
A) 网状模型、链状模型和层次模型      B) 层次模型、环状模型和关系模型  
C) 层次模型、网状模型和关系模型      D) 层次模型、网状模型和分布模型
- 2.关系数据库用表格数据来表示实体之间的联系，其三种关系操作是 ( )。  
A) 选择、投影和扫描      B) 选择、投影和连接  
C) 选择、运算和投影      D) 选择、投影和比较
- 3.关系模型数据库管理系统中存储与管理数据的基本形式是 ( )。  
A) 数据链      B) 关系树      C) 二维表      D) 流程图
4. ( ) 是存储在计算机内部的有结构的数据集合。  
A) 网络系统      B) 操作系统      C) 数据库      D) 数据库管理系统
- 5.数据库系统的核心是 ( )  
A) 数据库      B) 操作系统      C) 数据库管理系统      D) 编译系统
- 6.数据库管理系统按照被管理的 ( ) 分为三种，Visual FoxPro 属于 ( )。  
A) 名称      B) 使用方法      C) 数据模型      D) 大小  
E) 单用户 DBMS      F) 网络型 DBMS      G) 关系型 DBMS      H) 层次型 DBMS
- 7.说法 ( ) 是不正确的。  
A) 数据库减少了数据冗余  
B) 数据库避免了一切数据重复  
C) 数据库中的数据可以共享  
D) 若冗余是可控的，则系统可确保更新时的一致性
- 8.用二维表数据来表示实体之间联系的模型叫做 ( )。  
A) 网状模型      B) 层次模型      C) 关系模型      D) 实体-联系模型
- 9.用 DBMS 进行工资管理是计算机在 ( ) 方面的应用。  
A) 科学计算      B) 数据处理      C) 辅助设计      D) 人工智能

## 二、填空题

1. 计算机数据技术的发展经历了人工(程序)管理、文件管理和（ ）三个阶段。
2. DBMS 是（ ）的简称。
3. 关系数据库中的选择关系操作包括（ ）筛选和（ ）筛选。

## 第二章 Visual FoxPro 概述

### 2.1 Visual FoxPro 的特点

微机关系数据库管理系统在国内流行已久，并为众多用户所熟悉。关系数据库管理系统的每一步发展，都为用户所关注。目前，微机关系数据库管理系统已由最初的 dBASE 经 FoxBASE、FoxPro 发展到 Visual FoxPro，功能日益强大，操作更加灵活。1995 年，Microsoft 公司推出的 Visual FoxPro3.0，可以说是微机关系数据库发展史上的一个里程碑。它在技术内容、显示界面和程序设计思想方面都有着重要的改变和改进。Visual FoxPro3.0 使用了面向对象(Object Oriented)的编程技术和可视化(Visual) 的编程技术，是一个 32 位全新的 PC 平台关系数据库开发系统。1996 年，又推出的 Visual FoxPro5.0，它大大增强了 Internet 功能。目前 Visual FoxPro 的最高版本是 1998 年推出的 Visual FoxPro6.0，亦称 Visual FoxPro 98，它除继承和保持 Visual FoxPro 系列产品的特点和优点外，并使其成为一个更优秀的基于微机平台的关系型数据库管理系统。是目前最快、最完美的桌面数据库管理系统。

到目前为止，从 XBase 到 Visual FoxPro6.0，其发展趋势是功能不断增强、界面日趋友好，更容易学习和掌握。用户可通过 Visual FoxPro 的开发环境方便地设计数据库结构，管理表和数据库，设计应用程序界面，设计查询、报表、菜单等，并可利用项目管理器对数据库和程序进行管理，生成可执行文件，并进行发布等。本书基于 Visual FoxPro6.0 版本，介绍其基本原理及其使用方法。

Visual FoxPro 6.0 与以前的版本相比，其主要特点是：

#### 一、可视化编程技术

Visual FoxPro 6.0 与 XBase 和 FoxPro 相比的最大特点是在继承了以往数据库语言众多优点的基础上，引入了面向对象的编程思想，支持可视化编程技术，是数据库领域的一大进步。它普遍采用图形界面，提供了用于应用开发的各种向导、设计器、工具栏、菜单和生成器，操作简便、快捷、直观。这些设计工具不仅减轻了编程人员的工作量，而且给人一种“所见即所得”的感受。在您编辑表单、报表、菜单时，不必编程或很少编程即可实现美观实用的应用程序界面，且可以直接运行。可视化编程技术为不具备应用程序设计技术的广大用户，获得了开发应用程序的能力，也促使 Visual FoxPro 6.0 的推广应用。

#### 二、面向对象的编程思想

早期的 XBase 数据库程序设计语言只支持面向过程（结构化）的程序设计，面向过程的设计要求程序试图使现实事件的问题适合程序设计语言，因而要求程序设计人员必须有较高的程序设计技巧，采用该方法设计的数据库应用程序很难实现多任务操作。Visual FoxPro 在支持面向过程的编程方式的同时，还提供了面向对象的程序设计能力。面向对象的程序设计力图使程序设计环境适合现实世界，允许用户对“对象”（object）和“类”（class）进行定

义，并编写相应代码。Visual FoxPro 提供了两种类型近 30 个基类，用户可以在基类的基础上定义自己的类和子类。在 Visual FoxPro 的对象模式下，用户可以充分利用面向对象由事件驱动的应用程序设计方法，高效率地开发数据库应用软件。利用该方法开发的系统，不仅设计简单、易于掌握，而且用户界面美观、操作灵活，开发效率高。是逐步取代以往数据库语言的、正日渐普及的数据库编程语言。

### 三、真正意义的数据库概念

以前的 XBase 软件，均把数据库与一个二维表直接联系，使人容易产生一个数据库就是一个二维表的概念，而且每个数据库文件都是独立存在的。多数据库文件之间的联系，是由用户在程序中通过命令来描述。Visual FoxPro 废除了 XBase 不合理的数据库概念，引入了“数据库表”（DataBase Table）的新概念。数据库（DataBase）实际上是一个关于某一主题或目标的信息的集合。在 Visual FoxPro 中，数据库是一个容器（Container），用于管理存放在其中的对象。这些对象包括数据库表、视图（View）、关系（Relation）、存储过程（Stored Procedure）和连接（Connection）等。用户管理的二维表数据是存放在以文件形式存在的“表”中。Visual FoxPro 中的表分为两类：分属于某一数据库的数据库表和不属于任何数据库的自由表（free Table）。对所有数据库表，在建立表的同时，就定义了与数据库内其他表之间的关系。数据库实际是若干个表、表之间的关系和触发程序的集合，合理地体现了关系型数据库的思想，与关系型数据库理论统一了起来。新的数据库把有关系的表封装在一起，关系清晰、合理，数据处理方便，是真正意义的数据库概念。

### 四、增强了项目及数据库管理功能

Visual FoxPro 6.0 提供的另一高效工具是“项目管理器”，通过项目管理器，用户可以集中管理数据、文档、类库和源代码等各种资源。如建立更新数据库，设计或改变窗体和报表，定义或改变类库，生成或更改自己的应用程序等。用户在项目管理器中也可以使用向导、生成器和工具栏等高效的开发工具，使得项目管理工作变得非常容易。此外，其数据库的管理功能也十分强大，提供了一些大型数据库管理系统的功能，如设置表字段的默认值、字段及记录的有效性规则及表间记录的参照完整性规则等，极大地提高了数据的安全性。

### 五、通过对象链接与嵌入实现应用集成

“对象链接与嵌入”（OLE）是美国微软公司开发的一项重要技术。通过该技术，用户可以在 Visual FoxPro 环境下，共享 Word、Excel 等 Microsoft Office 应用软件中的数据。例如，在 Visual FoxPro 的表单（或窗体）中可以链接其他软件中的对象，并可对对象进行编辑。通过格式转换，用户可以在 Visual FoxPro 与其他软件之间进行数据输入与输出。用户甚至还可借助 OLE 中的控制，通过编程来运行其他软件，实现应用的集成。

### 六、支持网络应用

Visual FoxPro 6.0 既可单机应用，又适用于网络环境。所提供的视图和表单，不仅可访问和更新本地计算机中的数据，也可访问和更新远程服务器中的数据。该应用给用户提供了开发灵活的客户机/服务器应用程序的坚实的基础。

Visual FoxPro 6.0 拥有近 500 条命令，200 余种函数，其功能十分强大。由于采用了 Rushmore 快速查询技术，查询效率高；并扩大了对 SQL（关系数据库的标准语言）的支持，从而成为用户首选的数据库软件。

## 2.2 Visual FoxPro 运行环境

Microsoft Visual FoxPro 6.0 是一个 32 位的数据库开发系统,可运行于 Windows 95/98(中文版)或更高版本,或者 Windows NT 3.5 及其以上版本。在 Windows 95 / 98(中文版)操作系统中,运行 Visual FoxPro 6.0 的最低硬件配置是:

- l 中央处理器为 486(或更高档处理器),主频在 50MHz 以上处理能力的 IBM 兼容机。
  - l 16MB 以上内存容量(已设虚拟内存的计算机,要求内存容量在 8MB 以上)。
  - l 安装 Visual FoxPro 需要 90MB 的硬盘空间;安装后,硬盘上需要不少于 15MB 的自由空间。
  - l VGA 或更高分辨率的显示器。
  - l 鼠标、键盘、光驱等。
- 如果是多用户使用,还应具备相应的硬、软件网络环境。

## 2.3 Visual FoxPro 的启动与退出

成功地安装 Visual FoxPro 6.0 后,在使用之前要启动 Visual FoxPro 6.0,用后要关闭退出。

### 2.3.1 启动 Visual FoxPro 6.0

启动 Visual FoxPro 6.0 与启动 Windows 应用程序的方法相同,步骤如下:

打开屏幕左下角的“开始”菜单,单击“程序”选项,然后指向 Microsoft Visual FoxPro 6.0 组中的 Visual FoxPro 选项后单击,即可启动 Visual FoxPro 6.0。

如果是首次启动 Visual FoxPro 6.0,屏幕上会出现 Visual FoxPro 6.0 的操作选择对话框,该框用于选择启动后直接进入的操作。此时若选定该对话框中的“以后不再显示此屏”选项,则以后启动 Visual FoxPro 时将不再显示此屏。启动 Visual FoxPro 6.0 后,在不显示操作选择对话框的情况下,屏幕上会出现一个命令窗口,并在其中显示光标,标记启动成功。此后可以接受并执行 Visual FoxPro 6.0 的任何命令或操作。

### 2.3.2 退出 Visual FoxPro 6.0

退出 Visual FoxPro 6.0 的方法很多,使用下列四种方法之一,均可退出 Visual FoxPro 6.0,返回 Windows 界面。

方法一:在命令窗口中执行 QUIT 命令。

方法二:单击 Visual FoxPro 窗口标题栏右端的关闭按钮。

方法三:在 Visual FoxPro 窗口的“文件”下拉菜单中,单击“退出”选项。

方法四:直接按快捷键 Alt+F4。