

主 编 李 广
副主编 魏霖静
燕振刚
李 强



兰州大学出版社

Visual FoxPro

程序设计教程

Visual FoxPro

Visual FoxPro

Visual FoxPro

Visual FoxPro 程序设计教程

主 编 李 广

副主编 魏霖静 燕振刚 李 强



兰州大学出版社

图书在版编目(CIP)数据

Visual FoxPro 程序设计教程/李广主编. —兰州:兰州大学出版社, 2009. 8
ISBN 978-7-311-03465-8

I. V… II. 李… III. 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. TP311.138

中国版本图书馆 CIP 数据核字(2009)第 153082 号

策划编辑 梁建萍
责任编辑 龚 静
封面设计 管军伟

书 名	Visual Foxpro 程序设计教程
主 编	李 广
副 主 编	魏霖静 燕振刚 李 强
出版发行	兰州大学出版社 (地址:兰州市天水南路 222 号 730000)
电 话	0931-8912613(总编办公室) 0931-8617156(营销中心) 0931-8914298(读者服务部)
网 址	http://www.onbook.com.cn
电子信箱	press@onbook.com.cn
印 刷	兰州德辉印刷有限责任公司
开 本	787×1092 1/16
印 张	13.25
字 数	362 千
版 次	2009 年 8 月第 1 版
印 次	2009 年 8 月第 1 次印刷
书 号	ISBN 978-7-311-03465-8
定 价	25.00 元

(图书若有破损、缺页、掉页可随时与本社联系)

内 容 简 介

本书系统地介绍了中文 Visual FoxPro 6.0 及其程序设计技术, 内容包括数据库基础知识、数据与数据运算、表的维护及基本应用、数据库的基本操作、查询和视图、关系数据库标准语言 SQL、报表和标签、Visual FoxPro 程序设计基础、表单设计与运行、面向对象的可视化程序设计基础、进一步设计表单、菜单设计、项目管理器等。

为了方便教学和读者上机操作练习, 作者还编写了《Visual FoxPro6.0 程序设计实验教程》一书, 作为本书配套的实验教材。本书内容丰富, 通俗易懂, 可作为高等院校各专业及计算机培训班的教材或教学参考书, 也适合具有中等以上文化程度的读者自学之用。

前 言

Visual FoxPro 6.0 是 Microsoft 公司为数据库结构和应用程序开发的, 与 Windows98/2000/XP 环境兼容的, 全 32 位面向对象的快速数据库应用系统。它采用了可视化、面向对象的程序设计方法, 大大简化了应用系统的开发过程; 提高了系统的模块性和紧凑性; 提供了大量的系统开发工具和向导工具, 这些可视化的设计工具免除了开发者编写大量程序代码的工作。Visual FoxPro 6.0 中文版是快速掌握面向对象的程序设计、运用计算机解决常用数据处理问题的最佳入门语言。为此我们编写了这本《Visual FoxPro 程序设计教程》。

全书内容根据教学大纲和全国计算机等级考试大纲要求, 充分考虑了适合在校大学生的培养目标和教学特点, 本着由浅入深、循序渐进的原则, 注重基本知识和基本技能的培养, 突出其应用性, 每一章根据实验内容给出大致的操作提示, 使学生有的放矢地掌握教学内容, 并能在学完本教材后解决一些实际问题。此外, 本书的另一特点是理论教学与实践教学紧密结合, 力求做到理论教学和实践教学的同步融合, 达到学以致用目的。

本书包括数据库基础知识、数据与数据运算、表的维护及基本应用、数据库的基本操作、查询和视图、关系数据库标准语言 SQL、报表和标签、Visual FoxPro 程序设计基础、表单设计与运行、面向对象的可视化程序设计基础、进一步设计表单、菜单设计、项目管理器共 13 章内容。

由于时间仓促及编者水平有限, 书中错误和不足之处在所难免, 恳请广大读者批评指正。

编 者
2009 年秋

目 录

第 1 章 数据库基础知识 /001

1.1 数据库系统的基本概念 /001

1.2 关系数据库 /005

1.3 Visual FoxPro 系统概述 /008

第 2 章 数据与数据运算 /012

2.1 Visual FoxPro 的数据类型 /012

2.2 内存变量、数组和常量 /014

2.3 表达式 /018

2.4 基本函数 /020

第 3 章 表的维护及基本应用 /027

3.1 Visual FoxPro 命令概述 /027

3.2 表的建立 /029

3.3 表文件的编辑 /035

3.4 建立表索引 /042

3.5 表记录的查找 /045

3.6 表中数值参数的统计 /046

3.7 其他文件操作命令 /049

第 4 章 数据库的基本操作 /055

4.1 数据库的建立 /055

4.2 数据库的高级应用 /056

4.3 设置表 /058

4.4 建立表间关系 /061

4.5 使用多个表 /064

第 5 章 查询和视图 /072

5.1 创建查询 /072

5.2 创建视图 /077

5.3 创建参数视图 /079

5.4 使用视图更新数据 /080

第 6 章 关系数据库标准语言 SQL /082

6.1 SQL 语言概述 /082

6.2 数据定义 /083

6.3 数据查询 /087

6.4 数据操纵 /098

第 7 章 报表和标签 /101

- 7.1 创建报表实现数据输出 /101
- 7.2 利用报表进行分组、统计 /109
- 7.3 创建一对多报表 /112
- 7.4 使用标签实现数据输出 /112

第 8 章 Visual FoxPro 程序设计基础 /115

- 8.1 VFP 程序——命令文件的建立和执行 /115
- 8.2 VFP 常用编程命令及函数 /115
- 8.3 顺序执行 /118
- 8.4 判断分支 /122
- 8.5 循环结构 /127
- 8.6 过程与用户自定义函数 /136

第 9 章 表单设计与运行 /155

- 9.1 创建简单表单 /155
- 9.2 使用控件增加表单功能 /159
- 9.3 设置表单的数据环境 /163
- 9.4 创建表单集 /164
- 9.5 单文档界面与多文档界面 /166

第 10 章 面向对象的可视化程序设计基础 /169

- 10.1 面向对象的程序设计 /169
- 10.2 面向对象的基本概念 /169
- 10.3 类和对象的创建 /172

第 11 章 进一步设计表单 /176

- 11.1 控件的基本操作 /176
- 11.2 表单控件属性 /178
- 11.3 表单设计举例 /186

第 12 章 菜单设计 /191

- 12.1 菜单简介 /191
- 12.2 使用菜单设计器创建菜单 /191
- 12.3 进一步设计菜单 /192
- 12.4 创建快捷菜单 /196
- 12.5 菜单设计举例 /196

第 13 章 项目管理器 /198

- 13.1 项目文件的建立与打开 /198
- 13.2 使用项目管理器 /199
- 13.3 创建主控程序 /201
- 13.4 连编项目 /202

参考文献 /204

第 1 章 数据库基础知识

Visual FoxPro 是数据库家族的最新成员,也是其前身 FoxPro 与可视化程序设计相结合的产物。本章简介数据库系统的基本概念,概述关系数据库,对 Visual FoxPro 6.0 的特点、组成、界面、工作方式与核心工具等知识作概要介绍,为读者学习后续各章打下基础。

1.1 数据库系统的基本概念

数据库技术是在 20 世纪 60 年代末兴起的一种数据管理技术。在数据库出现前,计算机用户就使用数据文件来存放数据。常用高级语言从早期的 Fortran 到今天的 C 语言,都支持使用数据文件。常见的数据文件的格式是,一个文件包含若干个“记录”,一个记录又包含若干个“数据项”,用户通过对文件的访问实现对记录的存取。通常称支持这种数据管理方式的软件为“文件管理系统”。随着社会信息量的迅速增长,计算机处理的数据量不断增加。文件管理系统采用的一次最多存取一个记录的访问方式,以及在不同文件之间缺乏相互联系的结构,越来越不能适应管理大量数据的需要。于是数据库管理系统便应运而生,并在 20 世纪 60 年代末期诞生了第一个商品化的数据库系统——美国 IBM 公司的 IMS 系统(Information Management System)。从文件管理到数据库管理,代表了两代不同的数据管理技术。今天,数据库管理已成为计算机信息管理的主要方式。

1.1.1 数据库系统的组成

数据库系统具有数据的结构化、数据共享、数据独立性、可控冗余度等四个特点。它是把有关计算机硬件、软件、数据和人员组合起来为用户提供信息服务的系统。因此数据库系统是由计算机系统、数据库及其描述机构、数据库管理系统和有关人员组成的,是由这几个方面组成的具有高度组织性的总体。

1. 计算机硬件

计算机硬件是数据库系统赖以存在的物质基础,包括主机、键盘、鼠标、显示器、存储设备和打印机等。

2. 计算机软件

数据库系统中的软件包括操作系统、数据库管理系统及数据库应用系统等。数据库管理系统是数据库系统的核心软件之一。目前较流行的数据库管理系统有 Oracle、Sybase、SQL Server、Access、Visual FoxPro 等。数据库应用系统(Data Base Application System,简称 DBAS)是指基于数据库的应用系统。一个 DBAS 通常由数据库和应用程序两部分组成,它们都需要在 DBMS 支持下开发。数据库应用系统分为两类:

(1)管理信息系统。这是面向机构内部业务和管理的数据库应用系统。例如,人事管理

系统、教学管理系统等。

(2)开放式信息服务系统。这是面向外部、提供动态信息查询功能,以满足不同信息需求的数据库应用系统。例如,大型综合科技信息系统、经济信息系统和专业的证券实时行情、商品信息系统等。

3.数据库

数据库中的数据是以文件的形式存储在存储介质上的,是由数据库管理系统进行统一管理和控制的,是数据库系统操作的对象和结果。

4.数据库系统的有关人员

数据库系统的有关人员主要有3类:终端用户、数据库应用系统开发人员和数据库管理员。

1.1.2 数据库的三级模式结构

数据库领域公认的标准结构是三级模式结构,包括外模式、模式和内模式。美国国家标准协会 ANSI(American National Standard Institute)的数据库管理系统研究小组于1978年提出了标准化的建议,将数据库结构分为三级:面向用户或应用程序员的用户级、面向建立和维护数据库人员的概念级、面向系统程序员的物理级。用户级对应外模式,概念级对应模式,物理级对应内模式。数据库的三级模式结构如图1-1所示。

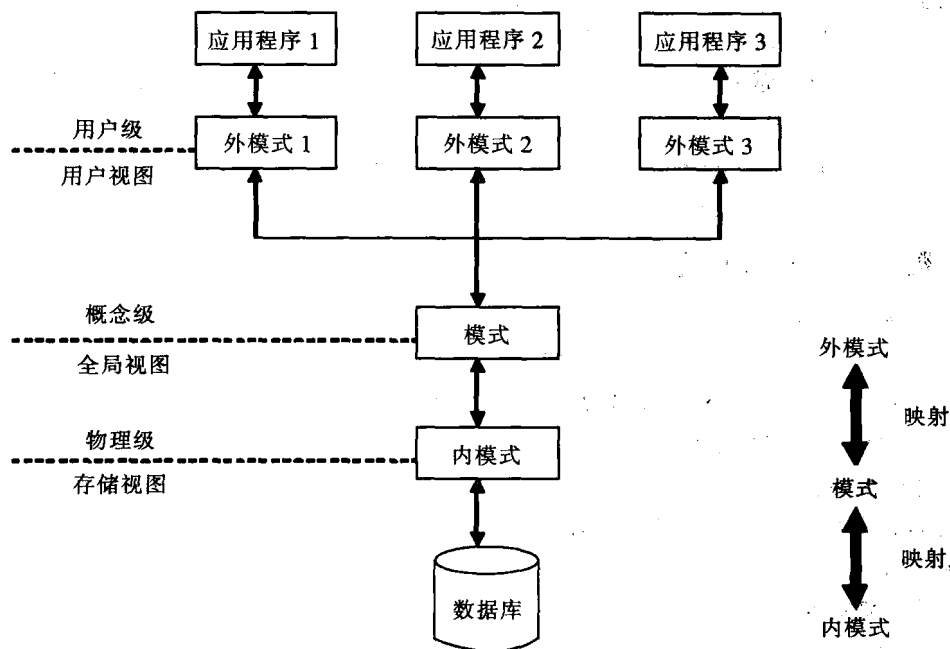


图 1-1 三级模式结构

1.模式

模式又称概念模式或逻辑模式,对应于概念级。它是由数据库设计者综合所有用户的数据,按照统一的观点构造的全局逻辑结构,是对数据库中全部数据的逻辑结构和特征的

总体描述,是所有用户的公共数据视图。

2. 外模式

外模式又称子模式,对应于用户级。它是某个或某几个用户所看到的数据库的数据视图,是与某一应用有关的数据的逻辑表示。

3. 内模式

内模式又称存储模式,对应于物理级。是数据库中全体数据的内部表示或底层描述,是数据库最低一级的逻辑描述,描述了数据在存储介质上的存储方式和物理结构,对应着实际存储在外存储介质上的数据库。内模式由内模式描述语言来描述、定义,是数据库的存储结构。

在一个数据库系统中,只有唯一的数据库,因而作为定义、描述数据库存储结构的内模式和定义、描述数据库逻辑结构的模式,也是唯一的,但建立在数据库系统之上的应用则是非常广泛、多样的,所以对应的外模式不是唯一的,也不可能是唯一的。

4. 三级模式间的映射

数据库的三级模式是数据在三个级别(层次)上的抽象。实际上,对于一个数据库系统而言,只有物理级数据库是客观存在的,是进行数据库操作的基础,概念级数据库不过是物理数据库的一种逻辑的、抽象的描述(即模式),用户级数据库则是用户与数据库的接口,是概念级数据库的一个子集(外模式)。

用户应用程序根据外模式进行数据操作,通过外模式—模式映射,定义和建立某个外模式与模式间的对应关系,将外模式与模式联系起来,当模式发生改变时,只要改变其映射,就可以使外模式保持不变,对应的应用程序也可保持不变;另一方面,通过模式—内模式映射,定义建立数据的逻辑结构(模式)与存储结构(内模式)间的对应关系,当数据的存储结构发生变化时,只需改变模式—内模式映射,就能保持模式不变,因此应用程序也可以保持不变。

1.1.3 数据模型

数据库中的数据是按一定的逻辑结构存放的,这种结构是用数据模型来表示的。数据模型是指数据库中数据与数据之间的关系,任何一个数据库管理系统都是基于某种数据模型的。数据模型是数据库系统中的一个关键概念,数据模型不同,相应的数据库系统就完全不同。目前,数据库管理系统常用的数据模型主要有3种:层次模型、网状模型和关系模型。

1. 层次模型

层次模型用树形结构来表示数据及其之间的联系,是数据库系统最早使用的一种模型,它的数据结构是一棵“有向树”。在层次模型中,各个数据对象之间是一对一或一对多的关系。层次模型的优点是结构简易、易于操作、向下寻找数据容易,但是层次模型不能直接表示多对多的关系,因而很难实现对复杂数据关系的描述。

如图1-2所示是一个层次模型,其中p1是根结点,p2、p3和p4是p1的子结点,因此它们也是兄弟结点,p3、p5、p6和p7是叶结点(没有子结点的结点,处于分枝的末端)。

层次模型的特点是:

- (1)有且仅有一个根结点,此结点无父结点,其层次最高。
- (2)其他结点有且只有一个父结点,但可以有若干个子结点。

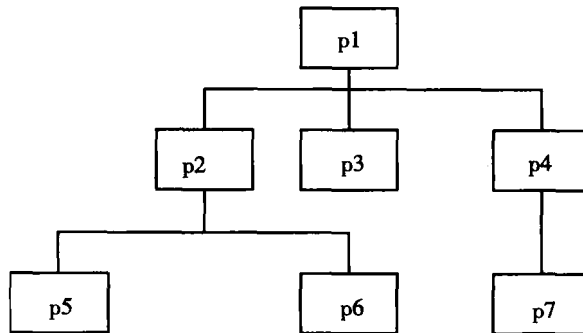


图 1-2 层次模型

2.网状模型

网状模型用网状结构表示数据及其之间的联系。在网状模型中,各个数据对象之间是一对一、一对多或多对多的关系,因而网状模型的结点之间可以任意发生联系,能够表示各种复杂的数据关系。网状模型是层次模型的拓展,网状模型结构比层次模型结构更有弹性,允许子结点有多个父结点,并且子结点之间的关系较接近,联系很容易。但其缺点是路径太多,当加入或删除数据时,牵动的相关数据很多,不易维护与重建。如图 1-3 所示是一个网状模型。网状模型的特点是:

- (1)一个结点可以有多个父结点。
- (2)可以有一个以上的结点无父结点。

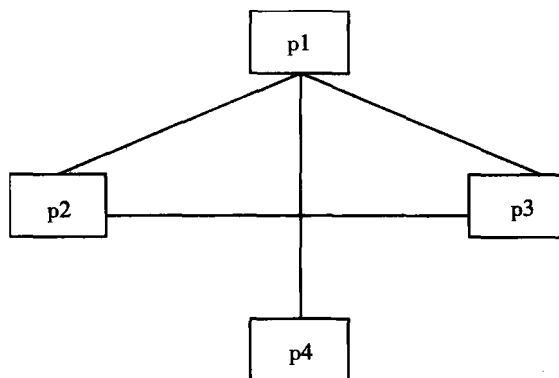


图 1-3 网状模型

3.关系模型

在数据库中把由行与列构成的二维表称为关系,关系模型是由关系表示的数据模型。关系模型是建立在数学概念基础上的,是一种易于理解并具有较强数据描述能力的数据库

模型。在关系模型中,数据在用户观点下的逻辑结构是一张二维表。关系模型是3种数据模型中最重要的模型,具有结构简单、清晰、易学易用等特点,是目前大多数数据库系统所采用的数据模型,并已成为数据库应用的主流。Visual FoxPro 就是一种典型的关系型数据库管理系统,如表 1-1 所示,就是一个关系模型的例子。

表 1-1 关系模型

学 号	姓 名	性 别	出生日期
200801001	李文清	女	11/23/89
200801002	徐丽丽	女	12/30/90
200801003	马圆圆	男	02/11/90
200801004	鲍美美	女	11/23/90

1.2 关系数据库

关系数据库是以关系模型建立的数据库,是若干个按照关系模型设计的表文件的集合。

1.2.1 关系的基本概念

1.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,其扩展名为 .dbf。

2.元组

二维表的每一行在关系中称为元组。在 Visual FoxPro 中,一个元组对应表中一个记录。

3.属性

二维表的每一列在关系中称为属性,每个属性都有一个属性名,属性值则是各个元组属性的取值。在 Visual FoxPro 中,一个属性对应表中一个字段,属性名对应字段名,属性值对应于各个记录的字段值。

4.域

属性的取值范围称为域。域作为属性值的集合,其类型与范围由属性的性质及其所表示的意义具体确定。同一属性只能在相同域中取值。

5.关键字

关系中能唯一区分、确定不同元组的属性或属性组合,称为该关系的一个关键字。单个属性组成的关键字称为单关键字,多个属性组合的关键字称为组合关键字。需要强调的是,关键字的属性值不能取“空值”。所谓空值就是“不知道”或“不确定”的值,因而空值无法唯一地区分、确定元组。

表 1-1 中“学号”属性可以作为单关键字,因为编号不允许重复。而“姓名”及“出生日期”等则不能作为关键字,因为学生中可能出现重名或出生日期相同的情况。但如果所有同名学生的出生日期不同,则可将“姓名”和“出生日期”组合成为组合关键字。

6. 关系模式

对关系的描述称为关系模式,其格式为:关系名(属性名 1,属性名 2,⋯,属性名 n)
关系既可以用二维表格来描述,也可以用数学形式的关系模式来描述。一个关系模式对应一个关系的结构。在 Visual FoxPro 中,也就是表的结构。

如表 1-1 对应的关系,其关系模式可以表示为:学生(学号,姓名,性别,出生日期)。

综上所述,一个关系就是一张二维表,由表结构和表记录组成,表的结构对应关系模式,表中的每一列对应关系模式的一个属性,每一列的数据类型及其取值范围就是该属性的域,所以,定义表就定义了对应的关系。而一个关系数据库由若干个关系组成,每个关系又是由关系模式确定的,每个关系模式又都包含若干个属性,因此,定义关系数据库时只需要定义每一个关系模式及每一个关系模式的属性。在 Visual FoxPro 系统中,与关系数据库对应的是数据库文件,一个数据库文件包含若干个表。

1.2.2 关系的基本特点

在关系模型中,关系具有以下基本特点:

(1)关系必须规范化,属性不可再分割。

规范化是指关系模型中的每个关系模式都必须满足一定的要求,最基本的要求是关系必须是一张二维表,每个属性值必须是不可分割的最小数据单元,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。因为该表的应发工资一列有三个子列,这与每个属性不可再分割的要求不符。只要去掉应发工资项,而将基本工资、奖金和津贴直接作为基本的数据项就可以了。

表 1-2 不能直接作为关系的表格示例

姓名	部门	职称	应发工资		
			基本工资	奖金	津贴
魏晓阳	信息学院	教授	600	245	1600
李文文	草业学院	讲师	450	200	900

(2)在同一关系中不允许出现相同的属性名。Visual FoxPro 不允许同一个表中有相同的字段名。

(3)关系中不允许有完全相同的元组。

(4)在同一关系中元组的次序无关紧要。也就是说,任意交换两行的位置并不影响数据的实际含义。

(5)在同一关系中属性的次序无关紧要。任意交换两列的位置也并不影响数据的实际含义,不会改变关系模式。

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。在这些基本要素中,有一点是关键,即属性不可再分割。

1.2.3 关系运算

在关系数据库中查询用户所需数据时,需要对关系进行一定的关系运算。关系运算主要有选择、投影和联接三种。

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。以逻辑表达式指定选择条件,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。

2.投影

投影运算是从关系中选取若干个属性的操作。投影运算从关系中选取若干属性形成一个新的关系,其关系模式中属性个数比原关系少,或者排列顺序不同,同时也可能减少某些元组。因为排除了一些属性后,特别是排除了原关系中关键字属性后,所选属性可能有相同值,出现相同的元组,而关系中必须排除相同元组,从而有可能减少某些元组。

3.联接

联接运算是将两个关系模式的若干属性拼接成一个新的关系模式的运算,对应的新关系中,包含满足联接条件的所有元组。联接过程是通过联接条件来控制的,联接运算从关系 R 和关系 S 的笛卡尔积 $R \times S$ 中,选取符合 $A \Theta B$ 条件的元组,即选择在关系 R 中 A 属性组上的值与在关系 S 中 B 属性组上的值满足比较操作 Θ 的元组。其中:A 和 B 分别为 R 和 S 上度数相等且可比的属性组, Θ 是比较运算符。设有关系 R 和 S,则 $R \times S$ 运算模式如图 1-4 所示。

关系 R		关系 S	
A	C	B	D
a1	c1	a1	d1
a2	c2	a3	d2

关系 $R \times S$			
A	C	B	D
a1	c1	a1	d1
a2	c2	a1	d1
a1	c1	a3	d2
a2	c2	a3	d2

图 1-4 笛卡尔积运算

1.3 Visual FoxPro 系统概述

1.3.1 系统配置要求

下面是在中文 Windows XP 中运行安装中文版 VFP6.0 的最低系统配置:

- 1.配有主频为 133MHz 或更高处理器的品牌机或兼容机(16MB 以上内存)。
- 2.VGA 或更高分辨率的监视器、Microsoft 兼容鼠标。
- 3.典型安装需要 100MB 硬盘空间,自定义安装需要 240MB 硬盘空间。

1.3.2 安装 VFP

在使用 VFP 前,首先必须把 VFP 系统软件全部安装到硬盘的某个子目录下。

1.安装 VFP6.0 中文版的准备工作

- (1)首先装入 Windows XP 操作系统。
- (2)如果计算机上使用了防病毒程序,“安装程序”将不能正常运行,请在运行“安装程序”之前将它关闭,运行“安装程序”后再打开。

2.安装 VFP6.0

可以直接从 CD-ROM 上安装 VFP6.0:

- (1)将 VFP6.0 安装盘插入 CD-ROM 驱动器中。
- (2)在光驱目录下(比如 F:盘)双击 setup.exe 文件,进入 VFP6.0 安装向导。
- (3)按照安装向导继续 VFP6.0 的安装,向导会提示用户 Microsoft 产品标志号,以及选择 VFP6.0 的安装类型。

对不同的用户,Microsoft 公司提供了两种不同的选项安装:

- ①典型安装:只安装最常用的组件。
- ②自定义安装:由用户决定安装的组件。对 VFP6.0 的初级用户,建议使用“典型安装”。

如果用户计算机的硬盘空间很大,可以选择“自定义安装”,然后单击“全部选中”。

默认情况下,VFP 的系统文件将被安装在“C:\Program Files\Microsoft Visual Studio\VFP98”目录下,用户可单击“更改文件夹”改变 VFP 的文件安装位置。

在上述各操作过程中,用户可以通过单击“上一步”按钮随时取消已确定的设置,或选择“退出”按钮退出 VFP 的安装,安装程序对系统不作任何修改。

当确定了每一项设置后,按“继续”按钮,安装程序开始向硬盘依次复制 VFP 程序。

(4)需要 VFP6.0 的帮助文档和示例的用户,还需要安装 MSDN(Microsoft Developer Network)。MSDN 的安装请按照其安装向导来完成,它需要 1.1GB 的硬盘空间。

(5)重新启动 Windows XP,完成 VFP6.0 的安装。

1.3.3 进入 VFP

在“开始”菜单的“程序”中双击“Microsoft Visual FoxPro 6.0”,可进入 VFP 的初始画面,画面上有五个命令按钮和一个选项,如图 1-5 所示。

其中“查看示例程序”,必须安装 MSDN。点击“关闭此屏”按钮,可进入 VFP 主窗口,如果希望以后不进入欢迎界面而直接进入 VFP 主窗口,选中“以后不再显示此屏”选项。

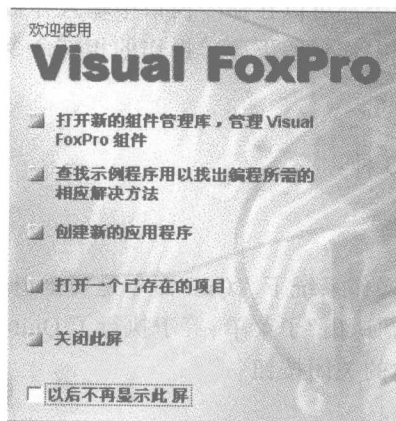


图 1-5 VFP6.0 的欢迎界面

VFP6.0 的主窗口由以下四部分组成:系统主菜单、常用工具条、命令窗口、工作区,如图 1-6 所示。

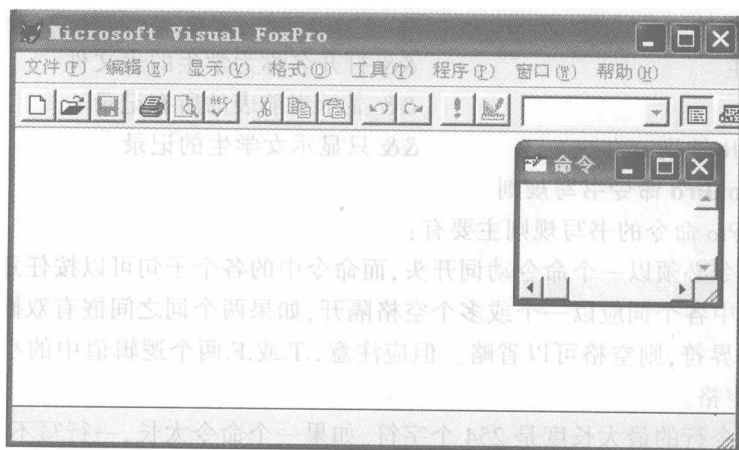


图 1-6 VFP6.0 的主窗口

1. 系统菜单

VFP 的菜单系统遵循 Windows 的惯例:系统主菜单条横跨在窗口顶部。主菜单项有文件、编辑、显示、格式、工具、程序、窗口、帮助等。当用户选择了某一菜单项,系统下拉显示出该菜单项的子菜单,用户可继续选择子菜单项。

2. 常用工具条

与其他 Windows 应用软件一样,VFP 提供给用户许多常用工具的快捷方法——快捷工具条。用户只需单击相应的按钮,就可以直接执行命令。

3.命令窗口

VFP 提供了两种操作方法:使用系统菜单或在命令窗口中输入命令。命令窗口是系统定义的窗口,VFP 中的所有命令都可以在命令窗口中输入并被执行。事实上,每次使用系统菜单进行操作,在命令窗口中均会留下菜单操作的相应命令。

1.3.4 退出 VFP

退出VFP 的方法有以下 3 种:

(1)在命令窗口中键入命令:

Quit

系统又退回到 Windows 操作系统下。Quit 命令是安全退出 VFP 的方法。

(2)选择“文件”菜单项的“退出”子菜单,等于执行了 Quit 命令。

(3)直接单击窗口右上角的关闭按钮。

1.3.5 VFP 的工作方式

1.命令格式

一般来说,VFP 的命令总是由一个称为命令字的动词开头,后随一个宾语和若干子句(称为命令子句),用来说明命令的操作对象、操作结果与操作条件。以下给出了若干简单的 VFP 命令的示例。

(1)USE 学生

&& 打开名称为学生的表文件

(2)LIST

&& 显示当前表的所有记录

(3)LIST FOR 性别="女"

&& 只显示女学生的记录

2.Visual FoxPro 命令书写规则

Visual FoxPro 命令的书写规则主要有:

(1)每个命令必须以一个命令动词开头,而命令中的各个子句可以按任意次序排列。

(2)命令行中各个词应以一个或多个空格隔开,如果两个词之间嵌有双撇号、单撇号、括号、逗号等分界符,则空格可以省略。但应注意,.T.或.F.两个逻辑值中的小圆点与字母之间不允许有空格。

(3)一个命令行的最大长度是 254 个字符。如果一个命令太长,一行写不下,可以使用续行符“;”,然后回车,在行末进行分行,并在下一行继续书写。即一个命令行可以分为若干个连续的物理行,其中除最后一个以外,各物理行应以分号结束。各物理行的长度之和不得超过 254 个字符。

(4)命令行的内容可以用英文字母的大写、小写或大小写混写。

(5)命令动词和子句中的短语可以用其前 4 个以上字母缩写表示,例如,DISPLAY STRUCTURE 可简写为 DISP STRU。

(6)一行只能写一条命令,每条命令的结束标志是回车键。

1.3.6 VFP 的两类工作方式

1.交互操作方式

从 DBase 到 FoxBase 的交互操作方式指的就是命令执行方式。用户只需记住命令的