

中文
版

Visual FoxPro

编程指南

6.0

李雨 主编

人民邮电出版社

386/cj

中文版 Visual FoxPro 6.0 编程指南

李雨 主编

人民邮电出版社

内 容 提 要

本书对中文版 Visual FoxPro 6.0 的使用方法、技巧以及编程技术进行了系统的阐述,并详细讲解了关系数据库技术,使读者能够较快较全面地掌握使用 Visual FoxPro 进行程序设计,尤其是数据库设计的方法。

本书首先对 Visual FoxPro 进行了概括的介绍,讲解了常用的几个程序设计工具,使读者能够尽快地进入编程状态,并可以运用这些工具检验后面讲到的概念和方法,然后集中篇幅讲解了关系数据库技术,尤其是其中的理论部分,使得用户能够科学地构建数据库管理系统;在初步掌握了数据库理论之后,我们又回到程序设计技术上来,全面详尽地讲解了 Visual FoxPro 的程序设计方法和思想。最后,还简要分析了 Visual FoxPro 系统提供的一个例子数据库系统,使读者对 Visual FoxPro 的程序结构有一个完整的把握。

在附录中,我们列举了一些常用的命令和函数,便于用户快速查询,并能够较快地掌握 Visual FoxPro 程序的主干。

作为一种强大的数据库设计工具,Visual FoxPro 所涵盖的内容是十分庞杂的。本书抓住其中的主要内容,进行了集中的讲解,这是为了满足大多数用户的需求。对于一些比较少用到的功能,在本书讲解的基础上,读者可以参阅 Visual FoxPro 的有关技术文件,从而解决问题。

本书语言浅显易懂,叙述全面,图文并茂,是读者学习 Visual FoxPro 的一本有益的参考书。

中文版 Visual FoxPro 6.0 编程指南

◆ 主 编 李 雨

责任编辑 滑 玉

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号

北京顺义向阳胶印厂印刷

新华书店总店北京发行所经销

◆ 开本:787×1092 1/16

印张:21.25

字数:523 千字

1999 年 9 月第 1 版

印数:6 001 - 11 000 册

2000 年 3 月北京第 2 次印刷

ISBN 7-115-08103-4/TP·1319

定价:33.00 元

编者的话

中文版 Visual FoxPro 6.0 是 Microsoft 公司推出的最新 FoxPro 版本,它集中了 Microsoft 公司的 Wizard 和 Rushmore 等最新技术,成为当今关系数据库市场上最重要的产品之一。

Visual FoxPro 6.0 被完整地合并到 Microsoft Visual Studio 之中,在 Windows 和 FoxPro 的基础上,又具有 Visual 系列的功能强大、直观易用、面向对象等特点,提供了 Wizard、Designer、Builder 等工具,使得数据库的设计和管理工作变得十分容易。

在 Visual FoxPro 6.0 中,增加了很多诸如对 GIF 和 JPEG 的支持、HTML 帮助、改进的向导数据库和设计器、对 2000 年的支持等新的特性,这些使得 Visual FoxPro 在原有的基础上更加符合用户的需求,使得用户能够快速开发数据库管理系统。在 Visual FoxPro 的集成开发环境中,可以使用向导快速地生成管理系统中常用的对象和控件,提高了程序开发的速度;当向导不能满足精细设计要求时,还可以使用设计器来对这些对象进行进一步的改进。针对每一种对象,比如数据表、表单、菜单等,Visual FoxPro 都提供了设计器,这些图形化的开发工具,使得用户能够根据实际需要形象地设计自己所需要的程序和界面,而避免了抽象的设计过程,提高了程序设计的效率,减轻了工作的强度。另外,为适应当今面向对象程序设计的潮流,Visual FoxPro 提供了对对象的支持。用户甚至可以独立地开发出自己的对象库系统,并可以积累这些工作的成果,提高代码的可维护性和可移植性。

Visual FoxPro 还可以与其它的应用程序方便地共享数据,也可以方便地与其它应用程序交换数据。对客户/服务器结构的支持,使得 Visual FoxPro 可以作为前台设计系统,与后台的 SQL Server 的数据库管理系统通过标准 SQL 命令紧密结合,方便地访问服务器,给用户带来了极大的方便。

本书作为一本使用和程序设计指南,目的在于帮助读者尽快掌握 Visual FoxPro 的使用方法、技巧和编程技术。本书的各章从各个角度,介绍了从基本的数据库理论到 Visual FoxPro 的使用方法等各个方面的内容,同时提供了简要的命令和函数检索,方便用户在开发和使用过程中能够快速找到并掌握各个语言部分的功能的使用方法。在整本书的编写过程中,还加入了大量的实例,便于读者边学习边实践,从而能够更好地掌握这门语言。

参加本书编写的,还有黄颖等。

由于时间仓促,不当之处在所难免,希望广大读者批评指正。

编 者

1999 年 8 月

目 录

第一章 Visual FoxPro 概览.....	1
1.1 Visual FoxPro 发展简介	1
1.2 关系数据库管理系统 (RDBMS) 概述	2
1.3 Visual FoxPro 的安装	3
1.4 Visual FoxPro 集成开发环境简介	4
1.5 小结	6
习题	7
第二章 数据表向导和设计器的使用	7
2.1 数据表向导的使用	7
2.1.1 生成数据表的第一步: 选定字段	9
2.1.2 生成数据表的第二步: 选择数据表的类型	10
2.1.3 生成数据表的第三步: 修改字段设置	11
2.1.4 生成数据表的第四步: 设置数据表索引	12
2.1.5 生成数据表的第五步: 选择当前对数据表的使用方法	12
2.2 数据表设计器的使用	13
2.2.1 字段选项卡	15
2.2.2 索引选项卡	20
2.2.3 “表” 选项卡	22
2.3 表达式生成器的使用方法	23
2.3.1 “表达式生成器” 对话框的使用方法	24
2.3.2 “表达式生成器选项” 对话框的使用	28
2.4 数据表的设计	29
2.5 输入和修改数据	30
2.5.1 输入数据	31
2.5.2 浏览数据	34
2.5.3 对数据表的其他操作	37
2.6 小结	40
习题	41
第三章 关系数据库理论初步	41
3.1 数据管理技术的发展历史	41
3.2 数据模型	42
3.3 关系数据库	43
3.3.1 概述	43
3.3.2 关系模型的基本概念	43

3.3.3 关系模型	45
3.4 小结	55
习题	56
第四章 查询与视图	57
4.1 查询的概念	58
4.2 查询向导的使用	58
4.2.1 用“查询向导”创建查询的第一步 - 字段选取	59
4.2.2 用“查询向导”创建查询的第二步 - 为表建立关系	60
4.2.3 用“查询向导”创建查询的第三步 - 包含记录	60
4.2.4 用“查询向导”创建查询的第四步 - 筛选记录	61
4.2.5 用“查询向导”创建查询的第五步 - 排序记录	62
4.2.6 用“查询向导”创建查询的第六步 - 限制记录	62
4.2.7 用“查询向导”创建查询的第七步 - 完成	63
4.3 查询设计器的使用	64
4.3.1 查询设计器	64
4.3.2 “查询设计器”工具栏	69
4.4 “查询向导选取”中的其它几种向导	70
4.4.1 本地视图向导	71
4.4.2 交叉表向导	71
4.4.3 图形向导	71
4.4.4 远程视图向导	72
4.5 视图设计器	74
4.5.1 视图设计器对话框	74
4.5.2 “视图设计器”工具栏	75
4.6 小结	76
习题	77
第五章 标准数据库操作语言 SQL 及其在 Visual FoxPro 中的实现	77
5.1 SQL 概述	78
5.1.1 一体化	78
5.1.2 SQL 的两种使用方式	79
5.1.3 一种高度非过程化的语言	79
5.1.4 简洁易学	79
5.2 SQL 数据定义功能	79
5.2.1 基本表的定义和修改	80
5.2.2 索引的建立和删除	96
5.2.3 视图的建立和删除	100
5.2.4 数据定义小结	103
5.3 SQL 数据操作功能	103
5.3.1 查询语句 SELECT	103

5.3.2 修改语句 UPDATE	110
5.3.3 删除语句 DELETE	112
5.3.4 插入语句 INSERT	113
5.4 SQL 数据控制功能	116
5.5 小结	117
习题	117
第六章 创建数据库	117
6.1 数据库向导	118
6.1.1 启动数据库向导	118
6.1.2 选择数据库模板	118
6.1.3 选择表和视图	120
6.1.4 为表建立索引	120
6.1.5 建立关系	120
6.1.6 完成	122
6.2 数据库的设计	122
6.2.1 确定数据库中包含的表	123
6.2.2 确定表的字段	124
6.2.3 确定主关键字	125
6.2.4 确定表间关系	126
6.2.5 存储过程的设计	127
6.2.6 触发器的设计	128
6.2.7 数据字典	128
6.3 数据库设计器	129
6.3.1 数据库设计器窗口	130
6.3.2 “数据库”菜单和数据库设计器工具栏	131
6.3.3 几个重要的工具部件	134
6.4 小结	139
习题	140
第七章 Visual FoxPro 6.0 程序设计语言	141
7.1 数据和字段类型	141
7.1.1 数值型 (Numeric)	142
7.1.2 字符型 (Character)	142
7.1.3 货币型 (Currency)	142
7.1.4 日期型 (Date)	142
7.1.5 日期时间型 (Date Time)	142
7.1.6 逻辑型 (Logic)	142
7.1.7 双精度型 (Double)	143
7.1.8 浮点型 (Float)	143
7.1.9 整型 (Integer)	143

7.1.10 备注型 (Memo)	143
7.1.11 通用型 (General)	143
7.1.12 字符型 (二进制)	143
7.1.13 备注型 (二进制)	143
7.2 数据存储	144
7.2.1 常量	144
7.2.2 变量	144
7.2.3 数组	144
7.2.4 字段	144
7.2.5 记录	145
7.2.6 对象	145
7.3 操作符	145
7.3.1 字符操作符	145
7.3.2 日期及时间操作符	145
7.3.3 逻辑操作符	146
7.3.4 关系操作符	146
7.3.5 算术操作符	146
7.4 表达式	146
7.4.1 字符表达式	147
7.4.2 日期表达式	147
7.4.3 逻辑表达式	147
7.4.4 关系表达式	147
7.4.5 名称表达式	148
7.5 常用函数	148
7.5.1 数学运算函数	148
7.5.2 字符操作函数	149
7.5.3 日期时间函数	150
7.5.4 类型转换函数	150
7.5.5 测试函数	151
7.6 程序流控制	152
7.6.1 条件分支语句	153
7.6.2 循环语句	153
7.7 过程和函数	154
7.7.1 过程和函数的基本形式	154
7.7.2 调用方法	154
7.7.3 变量的传递	155
7.8 程序设计的过程	155
7.9 小结	155
习题	156

第八章 表单设计	156
8.1 表单概述	157
8.2 表单向导	157
8.2.1 生成表单	158
8.2.2 生成一对多表单	163
8.3 表单设计器	166
8.3.1 使用快速表单来制作表单	167
8.3.2 “表单控件”工具栏	168
8.3.3 “表单设计器”工具栏	171
8.3.4 “表单”菜单	173
8.3.5 数据环境设计器	174
8.3.6 “代码”窗口	174
8.3.7 布局工具栏	176
8.3.8 自动格式生成器	176
8.4 小结	177
习题	178
第九章 控 件	178
9.1 控件概述	179
9.1.1 属性	180
9.1.2 事件	180
9.1.3 方法程序	181
9.2 显示编辑类控件	181
9.2.1 文本框控件	181
9.2.2 编辑框控件	184
9.3 选项类控件	187
9.3.1 选项按钮组控件	187
9.3.2 复选框控件	191
9.3.3 列表框控件	192
9.3.4 组合框控件	197
9.4 在给定的值域内接受用户输入的控件	199
9.4.1 在程序中限制用户的输入	199
9.4.2 微调控件	201
9.5 执行命令类控件	202
9.5.1 命令按钮控件	202
9.5.2 命令按钮组控件	203
9.6 显示类控件	204
9.6.1 图像控件	204
9.6.2 标签控件	205
9.6.3 形状控件和线条控件	205

9.6.4 页框控件	207
9.7 其它常用控件	209
9.7.1 表格控件	209
9.7.2 计时器控件	214
9.8 小结	215
习题	215
第十章 创建菜单和工具栏	215
10.1 规划菜单系统	216
10.2 应用程序主菜单的设计	216
10.3 创建子菜单	219
10.4 为菜单指定任务	220
10.4.1 为菜单或菜单项指定命令	220
10.4.2 使用过程完成任务	220
10.4.3 添加初始化和清理代码	221
10.5 生成并保存菜单文件	222
10.6 测试和调试菜单系统	223
10.7 创建自定义工具栏	223
10.7.1 定制 Visual FoxPro 工具栏	224
10.7.2 创建自定义工具栏类	226
10.7.3 在自定义工具栏类中添加对象	227
10.7.4 在表单集中添加自定义工具栏	227
10.7.5 为工具栏及其按钮定义操作。	228
10.8 协调菜单和自定义工具栏	228
10.9 小结	229
习题	229
第十一章 创建报表和标签	229
11.1 报表设计的步骤与方法	230
11.1.1 报表类型	230
11.1.2 报表设计的步骤	230
11.1.3 创建报表布局的方法	231
11.2 使用“报表向导”	231
11.2.1 字段选取	232
11.2.2 分组记录	232
11.2.3 选择报表样式	233
11.2.4 定义报表布局	234
11.2.5 排序记录	235
11.2.6 完成	235
11.3 使用报表设计器创建、修改与定制报表	236
11.3.1 报表设计器的组成	237

11.3.2 使用快速报表来创建报表	238
11.3.3 修改报表布局	239
11.4 设置报表数据源	244
11.5 预览和打印报表	245
11.5.1 预览报表	245
11.5.2 打印报表	245
11.5.3 打印技术	246
11.6 创建和使用标签文件	249
11.6.1 使用“标签向导”	250
11.6.2 标签设计器	251
11.7 向应用程序中添加报表和标签	251
11.7.1 使用报表和标签	252
11.7.2 集成查询和报表	252
11.8 小结	253
习题	253
第十二章 面向对象的程序设计	253
12.1 类和对象的概念	254
12.1.1 类的封装性	254
12.1.2 类的继承性	255
12.2 Visual FoxPro 中的面向对象程序设计概念	255
12.2.1 Visual FoxPro 类的分层结构	255
12.2.2 容器类与控件类	256
12.2.3 Visual FoxPro 中的对象	257
12.2.4 Visual FoxPro 的基类	258
12.2.5 Visual FoxPro 对象的属性	259
12.2.6 Visual FoxPro 对象的方法程序	259
12.2.7 在 Visual FoxPro 中使用对象	259
12.3 类设计器	261
12.3.1 打开类设计器	261
12.3.2 使用“类”菜单	263
12.4 类浏览器	270
12.4.1 “类浏览器”中的显示区域	271
12.4.2 窗口选项	271
12.4.3 “类浏览器”按钮	271
12.5 事件驱动的设计思想	272
12.5.1 事件的概念	272
12.5.2 Visual FoxPro 的核心事件集	272
12.5.3 事件的处理顺序	273
12.5.4 事件的代码	274

12.6 小结	274
习题	275
第十三章 建立应用程序	275
13.1 项目管理器	276
13.1.1 使用项目管理器	276
13.1.2 定制项目管理器	277
13.2 规划应用程序	278
13.2.1 用户需求分析	279
13.2.2 Tasmanian Traders 订单输入系统的设计	279
13.3 创建数据库	280
13.3.1 数据库的结构	280
13.3.2 使用存储过程	282
13.3.3 数据库中的自由表	282
13.4 设计操作界面和访问方法	282
13.4.1 应用程序的主菜单和工具栏	283
13.4.2 表单的设计	283
13.4.3 应用程序中的报表	285
13.5 面向对象设计方法的使用	287
13.5.1 工具栏类 ttoolbar	288
13.5.2 登录窗口类 login	289
13.6 编译应用程序	290
13.6.1 建立应用程序	290
13.6.2 连编应用程序	292
13.7 测试和调试应用程序	293
13.8 小结	294
习题	295
附 录	295
附录 A 数据类型函数	295
附录 B Visual FoxPro 环境设置命令	304
附录 C 程序设计命令	314
附录 D I/O 控制	316

第一章 Visual FoxPro 概览

1.1 Visual FoxPro 发展简介

Visual FoxPro 6.0 是 Microsoft 公司推出的最新 FoxPro 版本，它集中了 Microsoft 公司的 Wizard 和 Rushmore 等最新技术，成为当今关系数据库市场上最重要的产品之一。图 1-1 所示的为 Visual FoxPro 版本说明界面。

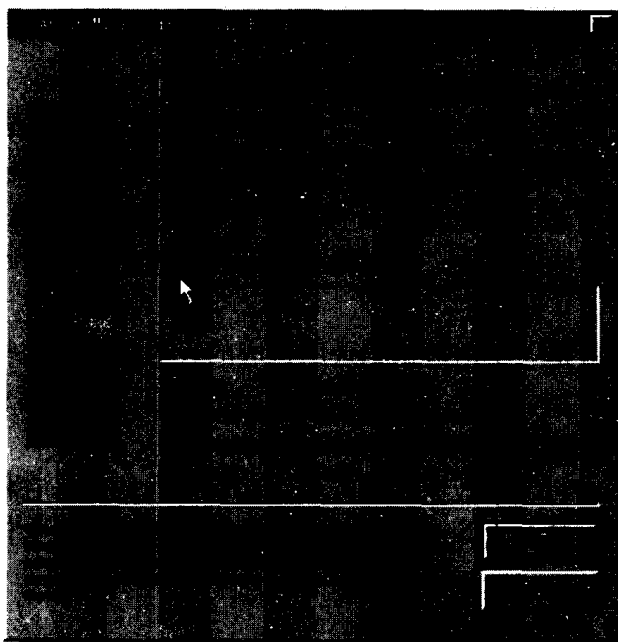


图 1-1 Visual FoxPro 的“关于”页面

早期的 PC 机数据库市场，是由 dBase 占据主要的支配地位。dBase 经历了从 dBase II 到 dBase IV 的演变过程，进行了大量的改进。同时，也有其他一些公司相继开发了很多与 dBase 兼容的数据库软件产品，并提供了更多的功能。这些产品，使得数据库的开发速度和程序设计能力得到了大大的提高，但是，一个不容忽视的问题，在于这些产品的数据库设计过程十分繁杂，因而难于使用。很多初学者，面对复杂的 dBase 语言，就知难而退，望而却步。甚至于程序开发人员，即使学会了使用这些开发工具，也由于其复杂的使用方法，而难于对语言本身有充分透彻的理解，或是将大量的精力与时间用于代码编制，而不能集中于数据库的设计工作，从而间接地影响了最终产品的质量。

Visual FoxPro 的推出，改变了这种被动的局面。它简便易用的界面，使得即使是普通的计算机用户，也可以轻松地完成数据库的设计工作，编制适合自己使用的小型数据库，赢得了广大用户的好评。同时在原有基础上不断推出的扩展功能，也受到了高级软件开发

人员的欢迎。

中文版 Visual FoxPro 6.0 是 Microsoft 公司开发的最新的 FoxPro 版本。它被完整地并到 Microsoft Visual Studio 之中, 在 Windows 和 FoxPro 的基础上, 又具有 Visual 系列的功能强大、直观易用、面向对象等特点, 提供了 Wizard、Designer、Builder 等工具, 使得数据库的设计和管理工作变得十分容易。

1.2 关系数据库管理系统 (RDBMS) 概述

数据库技术自 60 年代中期产生以来, 经历了几个时期。

1969 年, IBM 公司研制的数据库管理系统软件 IMS, 标志着层次型数据库的产生。

70 年代初, 美国数据系统语言协商会 CODASYL 下属的数据库任务组 DBTG 提出了 DBTG 报告, 确定并建立了网状结构数据库的概念、方法和技术, 成为数据库网状模型的基础和典型代表。

1970 年 IBM 公司的 San Jose 研究实验室的 E.F.Codd 发表了论文《大型共享数据库数据的关系模型》, 提出了数据库的关系模型, 开创了数据库关系方法和关系数据理论的研究, 为数据库技术奠定了理论基础。当今, 关系数据库理论得到了极大的发展和广泛的应用, 几种主要的数据库管理系统都是基于关系模型的。我们所使用的 FoxPro, 也是建立在这一模型之上。在后面的叙述中, 我们将看到, 这一基础给数据库的设计和实现带来了很大的便利。

简单地讲, 数据库就是一组由数据记录所构成的表的集合。这些记录又由字段组成: 一个字段属于一个数据类型, 代表了一种需要记录的信息。数据库管理系统 DBMS 就是用来管理和组织数据库中的数据记录的工具系统。

Visual FoxPro 是一套对关系数据库进行组织和管理的软件。它将数据表有效地结合起来, 完成一定的任务。但是, 它并不是简单地完成关系数据库的工作, 而是具有自己的特点:

● Visual FoxPro 更新了数据库的概念

传统的数据库模型, 是以二维表 DBF 作为一个完整的数据库, 每一个 DBF 自成一个数据库。而在 Visual FoxPro 中, 数据库的概念是若干个二维表、表间关系和触发器的集合。有关系的数据记录被封装在一起作为一个库, 没有关系的数据记录分属于不同的库, 不属于任何库的表称为自由表。可见, FoxPro 的关系是清晰且方便处理的。与此同时, Visual FoxPro 还定义了很多逻辑存在的数据子集, 以方便用户处理。

最后值得一提的是, Visual FoxPro 的数据库结构及操作与 SQL 等标准结构相一致, 这使得 Visual FoxPro 的数据交换和操作的实现更加标准、方便和合理。

● Visual FoxPro 的可视化编程方式

Visual FoxPro 继承了 Microsoft 的 Visual 系列产品的特色, 与 Visual C++、Visual J++、Visual Basic 等一样, 采用了可视化编程技术。这种可视化的编程方法, 降低了代码编制的复杂程度, 使得程序实现更加容易, 从而为程序设计人员带来了极大的方便, 使他们可以集中精力于软件的设计, 提高了软件的质量。

- 面向对象的程序设计思想

Visual FoxPro 不再采用以前结构化程序设计的方法，转而采用了面向对象的编程方法，降低了数据库和程序设计及维护的复杂程度。

- 对客户机/服务器结构的支持

Visual FoxPro 支持客户机/服务器结构，这使得用它设计编制的数据库能够应用于网络环境，适应于当前分布式系统的现实情况。Visual FoxPro 对这一结构的支持是十分重要的。

1.3 Visual FoxPro 的安装

首先，介绍一下如何安装 Visual FoxPro 系统，并通过配置使其满足用户个人的需要。

- 运行环境要求

Visual FoxPro 6.0 要求如下的软件运行环境：

Windows 95 以上的操作系统，或者 Windows NT 4.0 以上的操作系统。

Visual FoxPro 6.0 要求如下的硬件运行环境：

具有 486/66 以上 CPU，并与 IBM 兼容的微机；

16MB 或更多的内存；

对于典型安装，需要约 85MB 硬盘空间；对于完全安装，需要约 90MB 硬盘空间；

建议使用 VGA 或更高分辨率的显示器。

- 安装过程

可以通过光盘或网络来安装 Visual FoxPro 6.0。这里，我们介绍一下如何从光盘安装。需要提醒读者一点，就是如果您的计算机上正在运行反病毒软件，那么，在运行安装向导之前，请您将这些软件关掉，因为安装程序可能与这些软件冲突，从而不能正常运行。在安装完毕之后，务必将这些软件重新打开，以保护您的计算机系统。

安装 Visual FoxPro 6.0 的步骤如下：

- (1) 将安装光盘放入光盘驱动器；
- (2) 点击窗口左下角的“开始”菜单，并选择“运行”选项；
- (3) 在下拉列表框中输入“e:\setup”，并点击“确定”，其中，e:是光盘驱动器所在的驱动器号，应根据具体情况正确选择。
- (4) 以下只需根据安装向导的提示进行相应地选择即可。需要注意的只有以下三个选择：

如果需要最小安装（仅需要 15MB 硬盘），则在选择面板中选择定制安装，并去掉所有的选项，仅保留必需的文件。

如果需要典型安装（需要 85MB 硬盘），则在选择面板中选择完全安装，这一选择将安装包括 ODBC 在内的所有支持文件，而将帮助文件保留在光盘上。

当然，也可以采取完整安装。在选择面板中选择定制安装，并在选项对话框中选择全部，则会将所有文件安装到硬盘上。这需要 192MB 硬盘空间。

在安装过程中如果出现问题，请参阅随软件附带的用户手册，或向专家咨询。

安装完成之后，用户可能需要对系统进行适合自己需要的配置。以下介绍几个主要的配置环节：

1. 添加或删除 Visual FoxPro 组件

在安装过程完成之后，用户可能想添加一些遗漏的组件，或是删除一些没有必要安装的组件，以节省硬盘空间，这在定制安装中是十分常见的。可以进行如下操作以完成这一工作：

(1) 在 Windows 的操作面板中，选择添加/删除程序；

(2) 在弹出的对话框中选择 Microsoft Visual FoxPro 6.0，接着点击添加/删除按钮。

这之后，系统会弹出 Microsoft Visual FoxPro 6.0 Setup 对话框，在这个对话框中选择添加/删除按钮，就可以在定制窗口中通过选择或清除相应的选项来添加或删除相应的组件了。

2. 设置帮助系统

缺省情况下，帮助系统指向 MSDN。如果用户仅使用 Visual FoxPro，或是仅需要关于 FoxPro 的帮助，这一缺省设置显然是不够方便的。我们可以通过配置使 Visual FoxPro 的帮助直接指向有针对性的帮助系统。操作如下：

(1) 在工具菜单下，选择“选项”，接着选择文件位置选项卡。

(2) 在这一选项卡中，选择帮助文件栏目，并点击修改按钮。

1.4 Visual FoxPro 集成开发环境简介

在这一节，将向读者介绍一下 Visual FoxPro 的集成开发环境，以便于读者在后面的章节中一边学习，一边进行实践，真正掌握 Visual FoxPro 编程技术。

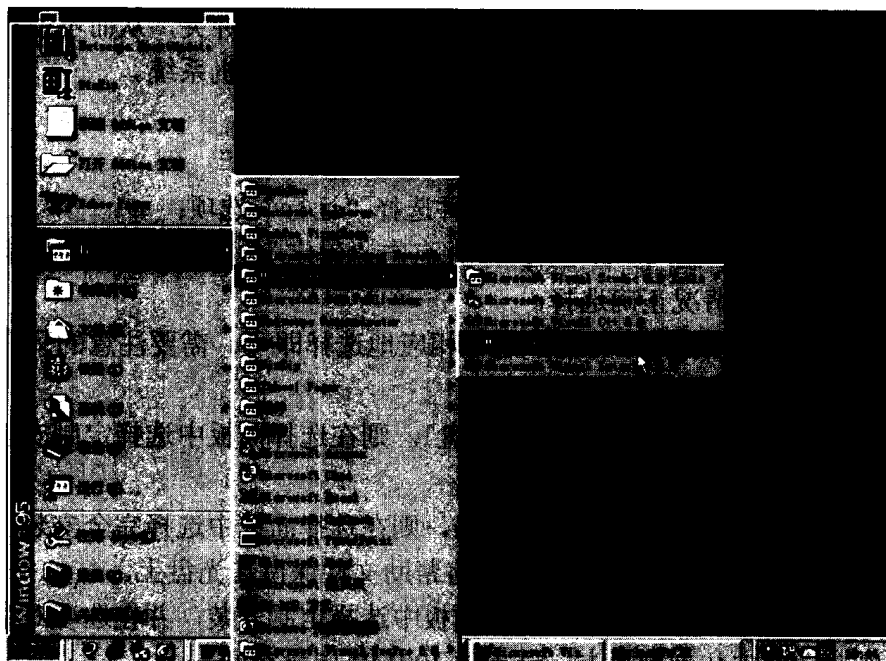


图 1-2 启动 Visual FoxPro

启动 Visual FoxPro 最简便的方法,是点击屏幕左下角的开始按钮,并选择相应的选项,如图 1-2 所示(不同的系统配置可能略有不同,但只需启动该程序项即可)。

Visual FoxPro 启动之后,如果是第一次使用,会弹出如图 1-3 所示的启动界面,只需关闭这个界面,就可以正式开始 Visual FoxPro 的使用了。



图 1-3 初次启动 Visual FoxPro 的启动画面

在 Visual FoxPro 的初始界面中,可以看到一个带菜单的窗口,以及窗口中一个命令子窗口。这就是我们使用的集成开发环境,如图 1-4 所示。在后面的讲解以及读者的使用中,将发现这一集成开发环境会给用户带来极大的方便,使程序开发工作变得简便易行。

集成开发环境界面由几部分组成:

名称条: 这是指窗口上方的蓝色条,它标志着现在所运行的是 Visual FoxPro 编辑工具,并将显示正在编辑的文件名。通过这一部分,我们可以对当前的工作状态有一个了解。

菜单条: 这是指名称条下方显示菜单的部分。多数情况下,我们的程序设计工作将从这里开始,比如新建和打开数据库、程序等等。

工具条: 这是主菜单条下方的部分。这里定义了许多快捷按钮,通过它们可以快速地启动某种功能。熟练地使用工具条,可以在一定程度上加快用户的操作速度。

在中心的工作区中,有一个小的窗口,我们称之为命令窗口。这是 Visual FoxPro 区别于其他数据库开发工具的一个优点。在命令窗口中,我们可以立即执行被键入的命令,不需要将其保存为文件并用程序方式执行。这种工作方式,便于程序员直接把握命令的执行结果,避免了冗长的编译运行时间。此外,在菜单和对话框中所作的选择可以马上转换为“命令”窗口中的命令。此时,程序员可以将这些命令直接复制并粘贴到 Visual FoxPro 程序中,然后在程序中使用这些代码,从而降低了编程的复杂程度。