



# Visual FoxPro 程序设计

主 编 薛庆文  
副主编 罗 峰 王秀巧 李 环 李振美

■ 高等院校人文素质教育课程规划教材

RENWEN SUZHI JIAOYU KECHENG



清华大学出版社

013063132

TP311.138FOX-43

193

高等院校人文素质教育课程规划教材

# Visual FoxPro 程序设计

主 编 薛庆文

副主编 罗 峰 王秀巧 李 环 李振美



TP311.138FOX-43

193

清华大学出版社

北 京



北航

C1671234

## 内 容 简 介

本书以 Visual FoxPro 6.0 简体中文版程序设计语言为背景, 详细介绍了 Visual FoxPro 6.0 程序语言的基础知识及在可视环境下进行可视化编程的方法与步骤。全书共分 8 章, 主要内容包括 Visual FoxPro 6.0 基础、数据库与表的基本操作、关系数据库标准语言 SQL、结构化程序设计、表单设计、报表与标签设计、菜单设计、开发实例——学生信息管理系统。

本书配套电子课件提供了教材所有例题和练习题的源代码, 以及教学用多媒体课件。

本书既可作为各类高等院校、各类高职院校非计算机专业学生学习 VFP 6.0 程序设计的教材, 也可作为广大计算机爱好者学习 VFP 6.0 程序设计语言的参考书。

本书封面贴有清华大学出版社防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

### 图书在版编目(CIP)数据

Visual FoxPro 程序设计/薛庆文主编. --北京: 清华大学出版社, 2013

高等院校人文素质教育课程规划教材

ISBN 978-7-302-32900-8

I. ①V… II. ①薛… III. ①关系数据库系统—程序设计—高等学校—教材 IV. ①TP311.138

中国版本图书馆 CIP 数据核字(2013)第 136577 号

责任编辑: 曹 坤

装帧设计: 杨玉兰

责任校对: 王 晖

责任印制: 刘海龙

出版发行: 清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址: 北京清华大学学研大厦 A 座 邮 编: 100084

社 总 机: 010-62770175 邮 购: 010-62786544

投稿与读者服务: 010-62776969, [c-service@tup.tsinghua.edu.cn](mailto:c-service@tup.tsinghua.edu.cn)

质 量 反 馈: 010-62772015, [zhiliang@tup.tsinghua.edu.cn](mailto:zhiliang@tup.tsinghua.edu.cn)

课 件 下 载: <http://www.tup.com.cn>, 010-62791865

印 装 者: 北京嘉实印刷有限公司

经 销: 全国新华书店

开 本: 185mm×260mm

印 张: 15.75

字 数: 377 千字

版 次: 2013 年 7 月第 1 版

印 次: 2013 年 7 月第 1 次印刷

印 数: 1~4000

定 价: 31.00 元

产品编号: 049255-01



目前,在普通高等院校的计算机基础教育中,计算机文化基础与计算机程序设计是两门前后承接的课程,联系非常紧密,计算机程序设计是计算机文化基础的后续课程。因此,选择简单明了的程序设计语言,有助于学生从简单的基础应用过渡到复杂的专业应用。

Visual FoxPro 6.0 是微软公司推出的关系数据库管理系统,是学习数据库技术的基础课程。它采用面向对象、事件驱动的编程方法,不仅提高了代码的可重用性,而且使程序的逻辑结构更加清晰,程序更加可靠和易于维护。另外,Visual FoxPro 提供了向导、生成器和设计器等三种工具,为快速高效地完成应用程序开发提供了强有力的支持。因此,在教学中选择了这门语言及其开发工具。

全书共分 8 章,主要内容包括 Visual FoxPro 6.0 基础、数据库与表的基本操作、关系数据库标准语言 SQL、结构化程序设计、表单设计、报表与标签设计、菜单设计等内容,最后通过设计一个具体的学生信息管理数据库系统,让学生掌握实际应用程序的开发技术。配套光盘提供了教材所有例题和练习题的源代码以及教学用多媒体课件,所有的程序均在中文 Visual FoxPro 6.0 企业版中调试通过。

全书内容安排合理、详略得当,讲述深入浅出,重点突出,避免了繁琐的理论阐述,突出了系统性和实践性。在各部分内容中均有解答详细、通俗易懂的例题,有利于学生加深对所学内容的理解。每章附有丰富的练习题,便于学生课后复习。全书建议授课时间为 72 学时(其中理论 36 课时和上机 36 课时)。

全书由薛庆文拟定教材编写提纲和配套光盘的结构及具体内容,并作最后统稿。全书具体编写分工如下:第 1、2 章由李振美编写,第 3、4 章由王秀巧编写,第 5、7 章由李环编写,第 6、8 章由罗峰编写。本书既可作为各类高等院校、各类高职院校非计算机专业学生学习 VFP 6.0 程序设计课程的教材,也可作为广大计算机爱好者学习 VFP 6.0 程序设计语言的参考书。

在本书的编写过程中,参考并引用了大量专家学者的著作、论文和网上资源,并得到了本单位领导和老师的大力支持和帮助,在此一并表示感谢。

虽然经多次讨论并反复修改,书中所有例子也已上机调试通过,但限于作者水平,书中难免有不当之处,敬请各位同行专家、读者给予批评指正。请本书的读者把对本书的意见和建议发邮件至 [xueqw88@163.com](mailto:xueqw88@163.com),我们会认真思考,以便再版时进行修改。

编 者



## 目 录

## 第 1 章 Visual FoxPro 6.0 基础.....1

|                                  |  |
|----------------------------------|--|
| 1.1 数据库系统概述.....1                |  |
| 1.1.1 数据及数据处理.....1              |  |
| 1.1.2 数据库系统的基本概念.....2           |  |
| 1.1.3 数据模型.....3                 |  |
| 1.2 Visual FoxPro 6.0 系统概述.....5 |  |
| 1.2.1 VFP 6.0 的安装、启动和退出.....5    |  |
| 1.2.2 中文 VFP 6.0 开发环境简介.....6    |  |
| 1.2.3 VFP 项目管理器.....9            |  |
| 1.2.4 VFP 的辅助工具.....11           |  |
| 1.3 VFP 6.0 的数据类型和数据存储.....12    |  |
| 1.3.1 VFP 的数据类型.....12           |  |
| 1.3.2 VFP 的数据存储.....13           |  |
| 1.4 运算符与表达式.....17               |  |
| 1.4.1 基本运算符与表达式.....17           |  |
| 1.4.2 宏替换.....20                 |  |
| 1.5 常用标准函数.....20                |  |
| 1.5.1 数值函数.....21                |  |
| 1.5.2 字符函数.....22                |  |
| 1.5.3 日期和时间函数.....23             |  |
| 1.5.4 数据类型转换函数.....24            |  |
| 1.5.5 测试函数.....25                |  |
| 本章小结.....27                      |  |
| 思考与练习.....28                     |  |

## 第 2 章 数据库与表的基本操作.....29

|                             |  |
|-----------------------------|--|
| 2.1 数据库的基本操作.....29         |  |
| 2.1.1 数据库和数据表的关系.....29     |  |
| 2.1.2 数据库的创建.....30         |  |
| 2.1.3 数据库的维护.....31         |  |
| 2.2 数据表的基本操作.....33         |  |
| 2.2.1 数据表结构的建立、显示和修改.....34 |  |
| 2.2.2 数据表的打开和关闭.....41      |  |

|                       |  |
|-----------------------|--|
| 2.2.3 记录的输入.....43    |  |
| 2.2.4 记录的浏览和显示.....44 |  |
| 2.2.5 记录的追加.....45    |  |
| 2.2.6 记录的修改.....48    |  |
| 2.2.7 记录的删除和恢复.....49 |  |
| 2.2.8 记录指针的定位.....51  |  |
| 2.3 数据表的高级操作.....52   |  |
| 2.3.1 记录的排序.....52    |  |
| 2.3.2 记录的索引.....53    |  |
| 2.4 数据表的统计计算.....58   |  |
| 2.4.1 求累计和及平均值.....59 |  |
| 2.4.2 统计记录个数.....59   |  |
| 2.5 多个数据表的操作.....59   |  |
| 2.5.1 工作区.....60      |  |
| 2.5.2 数据表之间的关联.....62 |  |
| 2.5.3 参照完整性.....66    |  |
| 本章小结.....67           |  |
| 思考与练习.....68          |  |

## 第 3 章 关系数据库标准语言 SQL.....69

|                            |  |
|----------------------------|--|
| 3.1 SQL 语言概述.....69        |  |
| 3.1.1 SQL 语言的基本概念.....69   |  |
| 3.1.2 SQL 数据定义功能.....70    |  |
| 3.1.3 SQL 数据操作功能.....73    |  |
| 3.1.4 SQL 查询命令.....75      |  |
| 3.2 查询.....79              |  |
| 3.2.1 利用查询设计器建立查询文件.....80 |  |
| 3.2.2 利用查询向导建立查询文件.....82  |  |
| 3.3 视图.....86              |  |
| 3.3.1 利用视图设计器建立视图.....86   |  |
| 3.3.2 利用视图向导建立视图.....89    |  |
| 3.3.3 利用 SQL 命令定义视图.....91 |  |
| 本章小结.....92                |  |

# Contents

|             |    |
|-------------|----|
| 思考与练习 ..... | 92 |
|-------------|----|

## 第4章 结构化程序设计 ..... 94

|                         |     |
|-------------------------|-----|
| 4.1 结构化程序设计基础 .....     | 94  |
| 4.1.1 结构化程序的概念 .....    | 94  |
| 4.1.2 结构化程序文件的建立 .....  | 96  |
| 4.1.3 结构化程序文件的执行 .....  | 97  |
| 4.2 顺序结构程序设计 .....      | 97  |
| 4.2.1 辅助语句 .....        | 97  |
| 4.2.2 输入命令 .....        | 98  |
| 4.2.3 格式输入/出命令 .....    | 100 |
| 4.2.4 顺序结构程序设计 .....    | 100 |
| 4.3 分支结构程序设计 .....      | 101 |
| 4.3.1 IF 分支语句 .....     | 101 |
| 4.3.2 IIF 函数 .....      | 102 |
| 4.3.3 IF 分支语句的嵌套 .....  | 103 |
| 4.3.4 多重分支语句 .....      | 103 |
| 4.4 循环结构程序设计 .....      | 104 |
| 4.4.1 FOR 循环 .....      | 105 |
| 4.4.2 DO WHILE 循环 ..... | 106 |
| 4.4.3 循环结构的嵌套 .....     | 107 |
| 4.4.4 SCAN 循环 .....     | 108 |
| 4.5 模块结构程序设计 .....      | 108 |
| 4.5.1 子程序及调用 .....      | 109 |
| 4.5.2 过程及其调用 .....      | 112 |
| 4.5.3 自定义函数 .....       | 113 |
| 本章小结 .....              | 115 |
| 思考与练习 .....             | 116 |

## 第5章 表单设计 ..... 117

|                         |     |
|-------------------------|-----|
| 5.1 面向对象的程序设计概述 .....   | 117 |
| 5.1.1 面向对象程序设计的概念 ..... | 117 |
| 5.1.2 VFP 6.0 中的类 ..... | 118 |
| 5.1.3 对象及引用 .....       | 120 |
| 5.1.4 事件及响应 .....       | 121 |
| 5.2 表单向导 .....          | 122 |
| 5.2.1 建立单表表单 .....      | 122 |
| 5.2.2 建立多表表单 .....      | 125 |

|                               |     |
|-------------------------------|-----|
| 5.3 表单设计器 .....               | 128 |
| 5.3.1 表单设计器的基本操作 .....        | 128 |
| 5.3.2 控件的操作与布局 .....          | 131 |
| 5.3.3 数据环境 .....              | 132 |
| 5.3.4 设置控件属性与编辑事件<br>代码 ..... | 134 |
| 5.3.5 表单的修改与运行 .....          | 136 |
| 5.4 常用表单控件 .....              | 136 |
| 5.4.1 输出类控件 .....             | 138 |
| 5.4.2 输入类控件 .....             | 140 |
| 5.4.3 控制类控件 .....             | 147 |
| 5.4.4 容器类控件 .....             | 153 |
| 5.4.5 连接类控件 .....             | 157 |
| 5.5 自定义属性与方法 .....            | 159 |
| 5.5.1 自定义属性 .....             | 160 |
| 5.5.2 自定义方法 .....             | 162 |
| 本章小结 .....                    | 165 |
| 思考与练习 .....                   | 165 |

## 第6章 报表与标签设计 ..... 167

|                               |     |
|-------------------------------|-----|
| 6.1 报表概述 .....                | 167 |
| 6.1.1 报表的结构 .....             | 167 |
| 6.1.2 报表控件 .....              | 168 |
| 6.1.3 报表预览与打印 .....           | 169 |
| 6.2 报表设计器 .....               | 171 |
| 6.2.1 报表设计器的启动 .....          | 171 |
| 6.2.2 报表设计器的使用 .....          | 172 |
| 6.2.3 使用报表设计器创建报表 .....       | 177 |
| 6.3 报表向导 .....                | 180 |
| 6.3.1 使用报表向导创建简单<br>报表 .....  | 181 |
| 6.3.2 使用报表向导创建一对多<br>报表 ..... | 185 |
| 6.4 快速报表 .....                | 188 |
| 6.5 标签 .....                  | 190 |
| 6.5.1 使用标签向导创建标签 .....        | 190 |
| 6.5.2 使用标签设计器创建标签 .....       | 192 |
| 本章小结 .....                    | 193 |

# Contents 目录



|             |     |
|-------------|-----|
| 思考与练习 ..... | 194 |
|-------------|-----|

## 第7章 菜单设计 .....

|                        |     |
|------------------------|-----|
| 7.1 规划菜单系统 .....       | 195 |
| 7.2 下拉式菜单 .....        | 196 |
| 7.2.1 创建下拉式菜单 .....    | 196 |
| 7.2.2 运行下拉式菜单 .....    | 201 |
| 7.2.3 在顶层表单中添加菜单 ..... | 205 |
| 7.3 快捷菜单 .....         | 206 |
| 7.4 快速菜单 .....         | 208 |
| 本章小结 .....             | 208 |
| 思考与练习 .....            | 209 |

## 第8章 开发实例——学生信息管理系统 .....

|                           |     |
|---------------------------|-----|
| 8.1 需求分析 .....            | 210 |
| 8.1.1 系统的主要需求 .....       | 210 |
| 8.1.2 学生信息管理系统的功能设计 ..... | 211 |
| 8.2 系统设计 .....            | 211 |

|                         |     |
|-------------------------|-----|
| 8.2.1 功能设计 .....        | 211 |
| 8.2.2 数据库设计 .....       | 212 |
| 8.3 系统实现 .....          | 214 |
| 8.3.1 建立项目 .....        | 214 |
| 8.3.2 创建主文件 .....       | 215 |
| 8.3.3 欢迎表单及登录表单设计 ..... | 216 |
| 8.3.4 系统主界面设计 .....     | 219 |
| 8.3.5 系统主菜单设计 .....     | 220 |
| 8.3.6 系统主要功能实现 .....    | 221 |
| 8.4 应用程序的调试与优化 .....    | 233 |
| 8.4.1 调试器的使用 .....      | 233 |
| 8.4.2 程序优化 .....        | 234 |
| 8.5 应用程序的发布 .....       | 235 |
| 8.5.1 编译与连编应用程序 .....   | 235 |
| 8.5.2 应用程序发布 .....      | 236 |
| 本章小结 .....              | 240 |
| 思考与练习 .....             | 240 |

## 主要参考文献 .....

# 第 1 章 Visual FoxPro 6.0 基础

## 学习目标

- 掌握数据库的概念
- 了解数据模型的分类、关系模型的概念和特点
- 掌握 VFP 6.0 的安装、启动及运行环境的配置方法
- 掌握 VFP 6.0 的基本数据类型、运算符及表达式
- 了解常用的基本函数

## 学习引导

Visual FoxPro 6.0 是 Microsoft 公司推出的 32 位数据库开发软件,用它来开发数据库,既简单又方便。Visual FoxPro 原名 FoxBase,是美国 Fox Software 公司推出的数据库产品,在 DOS 上运行,与 xBase 系列相容。FoxPro 原来是 FoxBase 的加强版,最高版本曾出过 2.6。Fox Software 被微软收购后,加以发展使其可以在 Windows 上运行,并且更名为 Visual FoxPro。

Microsoft Visual FoxPro 6.0 关系数据库系统是新一代小型数据库管理系统的杰出代表,它以强大的性能、完整而又丰富的工具、极高的处理速度、友好的界面以及完备的兼容性等特点,备受广大用户的欢迎。

Visual FoxPro 6.0 在操作方式上提供了菜单操作、命令操作和设计器操作三种方式,它们都能达到相同的效果。它还提供了一个集成化的系统开发环境,不仅支持过程式编程技术,而且在语言方面作了强大的扩充,支持面向对象可视化编程技术,并拥有功能强大的可视化程序设计工具。目前,Visual FoxPro 6.0 是用户收集信息、查询数据、创建集成数据库系统、进行实用系统开发较为理想的工具软件。

## 1.1 数据库系统概述

数据库是 20 世纪 60 年代末 70 年代初发展起来的一门新技术,它的出现使数据处理进入了一个崭新的时代。数据库的核心任务是数据管理,但是这并不意味着一开始就有数据库技术,数据库管理技术经历了由低级到高级、由简单到逐步完善的发展过程。

### 1.1.1 数据及数据处理

数据是存储在某一种媒体上能够识别的物理符号,通过对数据进行加工处理获取有价值的信息,可以对所需者的活动提供决策依据。



数据处理是指对数据进行一系列加工、存储、合并、分类、计算等操作的过程。随着计算机技术的发展,数据处理技术大致经历了人工管理、文件系统管理和数据库系统管理三个阶段。

### 1. 人工管理阶段

在 20 世纪 50 年代中期以前,由于没有必要的软、硬件环境的支持,用户对数据的处理只能在裸机上采用批处理方式。在这种管理方式下,用户的应用程序和数据之间互不独立,并且一一对应,也就是说,应用程序完全依赖于数据,数据结构的改变、数据存取的物理地址或存储设备的变化,都会要求修改相应的应用程序。

### 2. 文件系统管理阶段

20 世纪 50 年代后期至 60 年代后期,由于操作系统中有了专门用于管理数据的软件,这种软件被称为文件管理系统。在该阶段,把要处理的有关数据组织成文件,应用程序通过文件管理系统对数据文件中的数据进行加工处理,此时应用程序和数据之间虽然具有了一定的独立性,但是这种独立性是相当差的。如:一旦数据的结构改变了,就要修改应用程序;反之,一旦应用程序的结构改变,也必然引起数据结构的改变。

### 3. 数据库系统管理阶段

20 世纪 60 年代后期,需要处理的数据急剧增加,文件管理方式显示出不足,此时数据库管理技术便应运而生。数据库管理技术对海量数据实行统一管理,形成一个数据仓库,库中的所有数据供不同的用户共享,即应用程序和数据之间实现了完全独立。这样不仅使应用程序对数据的访问和管理方便灵活,而且使数据冗余大大减少。

## 1.1.2 数据库系统的基本概念

### 1. 数据库

数据库(DataBase, DB)是以一定方式组织在一起的结构化的相关数据的集合,存放在外存储器上,能够被多个用户所共享,通常把这样一个数据集合形象地称为数据库。

### 2. 数据库管理系统

数据库管理系统(DataBase Management System, DBMS)是对数据库进行管理和控制的软件系统,是数据库系统的中心枢纽。用户对数据库进行的各种操作——数据库的建立、使用和维护,都是在 DBMS 的统一管理和控制下进行的。Visual FoxPro 6.0 就是一个常用的数据库管理系统。

### 3. 数据库系统

数据库系统(DataBase System, DBS)是指计算机系统中引入数据库之后组成的系统,





它是用来组织和存取大量数据的系统。它由计算机系统(硬件和基本软件)、数据库、数据库管理系统和有关人员(数据库管理员、应用设计人员、最终用户)组成的具有高度组织性的总体。

### 1.1.3 数据模型

在现实世界中,事物之间的联系是客观存在的,这种联系是由事件本身的性质决定的。通常把表示客观事物及其联系的数据及其结构称为数据模型。

#### 1. 数据模型的基本概念

##### 1) 实体

客观存在并且可以相互区别的事物称为实体。

##### 2) 属性

实体所具有的某一特性称为属性。一个实体可以由若干个属性来描述。

##### 3) 码

唯一标识实体的属性或属性的组合称为码。在 Visual FoxPro 中对应的概念是关键字。

##### 4) 域

属性的取值范围称为域。不同的属性有不同的取值范围,即不同的域。

##### 5) 联系

实体之间的对应关系称为联系,它反映现实世界中事物之间的相互关系。实体之间的联系可以归纳为三类,即一对一的联系、一对多的联系和多对多的联系。

#### 2. 数据模型的分类

按照实体之间的不同联系方式,数据库分为三种数据模型,即层次模型、网状模型和关系模型。最常用的数据模型是关系模型。

##### 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。在层次模型中,数据被组成从“根”开始倒挂的“树”,每个实体被当成一个节点,由“根”开始沿不同的分支放到不同的层次上。如果某个实体不再向下分支,则该实体所对应的节点称为“树叶”。上级节点与下级节点之间为一对多的联系。事实上,许多实体间的联系本身就是自然的层次关系,如某学院的层次模型如图 1-1 所示。

采用层次模型设计的数据库管理系统称为层次数据库管理系统,在其中建立的数据库称为层次数据库。层次模型的数据库管理系统也是最早出现的数据库管理系统。

##### 2) 网状模型

网状数据模型用以实体为节点的有向图来表示实体之间的联系。在网状数据模型中,至少有一个节点有多余一个的父节点,也可以有一个以上的节点无父节点。如某计算机系的网状模型如图 1-2 所示。

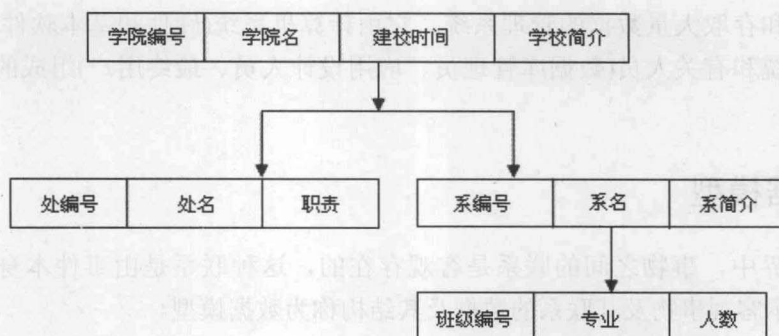


图 1-1 层次模型示意图

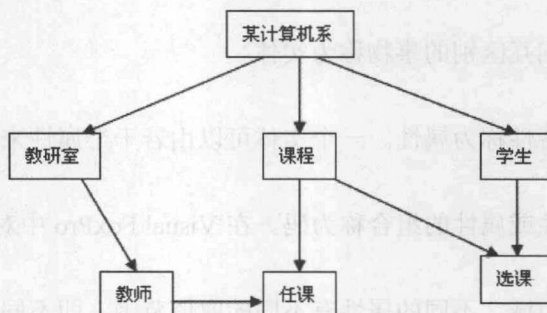


图 1-2 网状模型示意图

### 3) 关系模型

20 世纪 70 年代初, IBM 的研究员 E.F.Codd 博士首先提出了关系的概念, 一个关系的逻辑结构就是一个二维表。这种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模型。如一个学生关系表如表 1-1 所示。

表 1-1 学生关系表

| 学 号       | 姓 名 | 性 别 | 出生日期       |
|-----------|-----|-----|------------|
| 201203001 | 张丽丽 | 女   | 1995/01/19 |
| 201203002 | 李 道 | 男   | 1994/07/21 |
| 201203003 | 王英霞 | 女   | 1994/08/08 |

以关系模型为基础设计的数据库管理系统称为关系数据库管理系统, 在这种系统中建立的数据库称为关系数据库。虽然关系模型出现的比较晚, 但由于它是建立在严格的数学理论基础上的, 所以关系数据库管理系统也是当前最流行的数据库管理系统。如 Visual FoxPro、Paradox 等都是关系型数据库管理系统。在此, 以 Visual FoxPro 6.0 为例来介绍关系模型中的相关术语。

(1) 关系: 关系就是一个二维表, 每个关系都有一个关系名。在 VFP 6.0 中一个关系就是一个数据表(二维表), 每个数据表都存储为一个文件, 扩展名为.dbf, 简称为表, 如





表 1-1 所示。

(2) 元组：在一个关系中，水平方向上的每一行称为一个元组。在 VFP 6.0 的数据表中，每一行中称为记录，同一个数据表中的若干个记录的性质是相同的，格式、长度是相等的。

(3) 属性：在一个关系中，垂直方向上的列称为属性，属性是不可再分的数据单位，每一列有一个属性名。在 VFP 6.0 的数据表中，每一列称为一个字段，每个字段都有一个字段名。

(4) 域：是属性的取值范围，即同一个关系中不同的元组对同一个属性的取值所限定的范围。

(5) 主关键字：是属性或属性的组合，其值能够唯一地标识一个元组。

#### 4) 关系的完整性约束

关系的完整性约束是为了保证数据库中数据的正确性和兼容性对关系模型提出的某种约束条件或规则。完整性约束只有在创建数据库以及数据库更新操作时才需进行检查。关系的完整性约束大致可以分为以下几类。

##### (1) 域完整性约束

域完整性约束也称为用户自定义完整性约束，主要约束一个属性(字段)的取值范围、取值类型及有效性规则等。

##### (2) 实体完整性约束

为了确保实体完整性，每个关系中应有一个主关键字段，每个记录的主关键字段的值是唯一的，而且不能为空值。

##### (3) 参照完整性约束

参照完整性约束是对关系数据库中建立关联关系的数据表之间数据参照引用的约束，也就是对外部关键字的约束，如同步修改外部关键字或同步进行删除，这样可确保表之间的记录完整性。

总之，为了保持数据库中数据的一致性和正确性，必须使数据库中的数据满足完整性约束。

## 1.2 Visual FoxPro 6.0 系统概述

Visual FoxPro 6.0 简称为 VFP 6.0，是 Microsoft 公司开发的数据库开发软件。它提供了一个集成化的系统开发环境，不仅支持过程式编程技术，而且在语言方面作了强大的扩充，支持面向对象可视化编程技术，并拥有功能强大的可视化程序设计工具。它在操作方式上提供了菜单操作、命令操作和设计器操作三种方式，它们都能达到相同的效果。

### 1.2.1 VFP 6.0 的安装、启动和退出

VFP 6.0 可在 Windows 95/98(中文版)或 Windows NT 4.0(中文版)或更高版本中运行。



在 Windows 操作系统中运行 Visual FoxPro 的要求很低, 现在流行和使用的计算机都能够满足这些要求。

### 1. VFP 6.0 的安装

VFP 6.0 的安装和其他 Windows 平台下的软件安装方法基本一样, 可根据安装向导的提示一步一步进行安装即可。在这个过程中, 只需要输入极少量的信息和作出一些必要的选择。安装完成后, 需要对系统进行必要的配置, 以方便将来使用。在此, 不再详细介绍。

### 2. VFP 6.0 的启动

VFP 6.0 的安装好以后, VFP 6.0 的启动方法与 Windows 下的应用程序相同, 在此不再详述。

使用 VFP 6.0 完成数据处理以后, 需要退出 VFP 环境, 以避免数据的丢失。

### 3. VFP 6.0 的退出

以下几种操作方法均可退出 VFP 6.0 环境。

方法一: 从“文件”下拉菜单中选择“退出”选项。

方法二: 用鼠标单击 VFP 系统窗口右上角的“×”关闭窗口图标。

方法三: 在 VFP 的命令窗口中输入 Quit 命令, 再按 Enter 键。

方法四: 单击 VFP 主窗口左上方的狐狸图标, 从弹出的下拉菜单中选择“关闭”或按 Alt+F4 组合键。

## 1.2.2 中文 VFP 6.0 开发环境简介

### 1. 中文 VFP 6.0 的用户界面

VFP 6.0 启动后, 即可进入 VFP 6.0 主窗口, 它和其他的 Windows 窗口类似, VFP 6.0 主窗口主要由标题栏、菜单栏、工具栏、状态栏、命令窗口和工作区域等组成, 如图 1-3 所示。在此只简单介绍一下命令窗口和工作区。

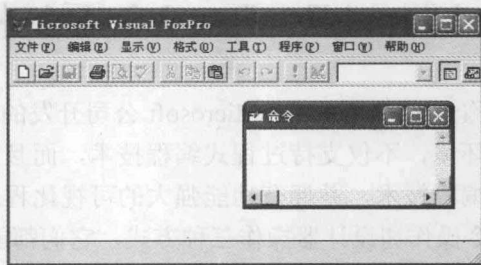


图 1-3 VFP 6.0 系统的用户界面

#### 1) 命令窗口

对于新用户来说, 通过菜单和对话框可以很容易地使用 VFP 的功能而不需记忆各种命





令。当然也可以通过使用命令窗口输入相应的命令来完成数据处理的操作，通常使用命令窗口的操作方法更为简便。

使用命令操作时，在命令窗口中输入 VFP 命令后，然后按 Enter 键即可立即执行相应的操作。在“窗口”菜单下，选择“隐藏”命令可以关闭命令窗口；再选择“命令窗口”命令可以重新显示命令窗口。

## 2) 工作区域

在工作区域可以展开系统中的各种窗口，使用各种向导，也可以直接在工作区域中输出结果。

## 2. VFP 运行环境的配置

在正确安装了 VFP 之后，通常需要对其运行环境进行配置，让用户使用起来更加符合个人的习惯。对 VFP 6.0 进行配置的方法可以有很多，可以利用系统菜单中“选项”对话框进行交互式设置，也可以采用编程或直接修改注册表文件的方法来订制自己的配置文件。

下面以菜单方式的环境配置为例来介绍 VFP 6.0 运行环境的配置方法。

启动 VFP 6.0 之后，选择“工具”菜单中的“选项”命令，即可以打开“选项”对话框，如图 1-4 所示。通过该对话框中相应的选项卡，可以方便地查看、更改 VFP 6.0 系统设置。

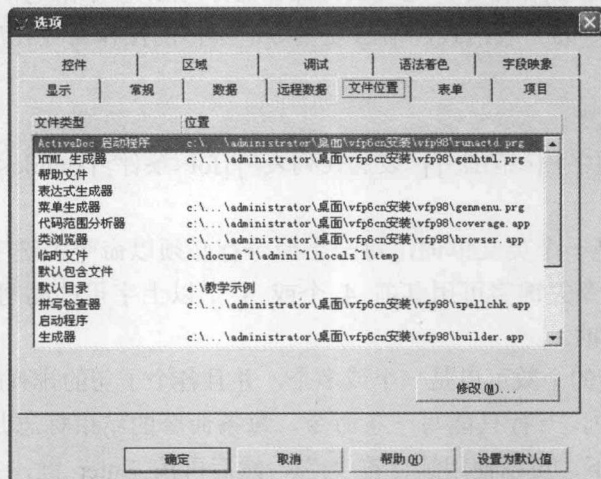


图 1-4 “选项”对话框

VFP 6.0 设置既可以临时修改，也可以是永久修改。临时的设置信息在退出 VFP 6.0 时从内存中释放，而永久设置在退出 VFP 6.0 时不失效，一直作为系统默认值，直到再次进行设置修改为止。在“选项”对话框中配置完毕后，直接单击“确定”按钮，则是临时配置，若先单击“设置为默认值”，再单击“确定”，则是永久配置。

**注意：**在未弄清各项意义之前应取其默认值，不要随便更改，以免系统出错。



### 3. VFP 的三种工作方式

#### 1) 菜单操作方式

用户通过选择菜单, 进入窗口或对话框来完成工作的工作方式称为 VFP 的菜单操作工作方式。每一次执行菜单命令, 在命令窗口中一般都会显示出与菜单操作对应的命令内容。

#### 2) 命令交互方式

根据需要, 采用人机交互方式在命令窗口中输入命令, VFP 通过执行这些命令来完成操作和数据处理。命令交互方式简捷迅速, 大多数的菜单操作都是与某个命令操作相对应。

#### 3) 程序执行方式

对于复杂的问题或较长的处理过程, 利用菜单方式或命令方式进行操作时, 就需要多次选择菜单或在命令窗口输入大量命令才能完成, 使用起来不但欠灵活而且效率也低, 这时最佳选择就是采用程序方式。即先在程序编辑窗口编写好程序, 然后再利用菜单或命令窗口执行该程序, 就能方便地进行各种数据处理和操作。

### 4. VFP 的基本操作命令

在 VFP 中, 有很多操作命令, 通过执行操作命令可以完成相关操作和数据处理。本章主要介绍基本操作命令的一般格式、语法规则等, 具体的操作命令在后面的章节中将逐步介绍。

#### 1) 命令的一般格式

格式: <命令关键字> [<范围>] [<表达式列表>] [for<条件>] [while<>]

说明:

① 命令关键字是一个英文单词, 每一个命令行必须以命令关键字开头, 字母大小写等效。对于较长的命令关键字可用其前 4 个或 4 个以上字母的缩写表示, 如 append、APPE、Appen 是等效的。

② 命令行中子句的个数可以是一个或多个, 并且各个子句的排列顺序是任意的。

③ 在命令窗口中, 一行只能写一条命令, 每条命令的结束标志是回车键。如果一个命令太长, 一行写不下, 可以使用续行符“;”, 然后再按 Enter 键, 并在下一行接着书写该命令。但总长不能超过 254 个字符。

④ 命令中的所有字符(除汉字外)一律在英文状态下输入。

#### 2) 命令格式中的符号约定

(1) <>: 表示必选项。该项必须根据具体问题选择一个确定的参数。

(2) []: 表示可选项。若选, 根据具体问题选择一个参数; 不选, 则取系统的默认值。

(3) |: 表示任选一项。在以该符号作分隔符的多个选项中任选一项(只能选一项)。

(4) ...: 表示重复出现项。在一个命令或函数表达式中, 某一部分可以按同一方式重复。

注意: 在实际输入命令时, 上述符号都不要输入。





### 3) 命令中的表达式列表子句

表达式列表是一个或多个由逗号隔开的表达式。表达式除了可以是字段名外,还可以是算数表达式、日期表达式等各种表达式。

### 4) 命令中的范围子句

在 VFP 命令中,常用<范围>子句来指定可以操作的有效记录的范围,一般有以下 4 种选择。

(1) ALL: 对当前表中所有记录进行操作。

(2) NEXT<n>: 指从当前记录开始的连续 n 条记录。

(3) RECORD<n>: 对记录号为 n 的记录进行操作。

(4) REST: 对从当前记录开始到表末尾的所有记录进行操作。

### 5) 命令中的条件子句

(1) FOR<条件表达式>: 对满足条件的记录进行操作。使用 FOR 子句,系统将记录指针重新指向表文件顶部(不选择“范围”短语,默认范围是 ALL 时)或将记录指针指向范围内的第一条记录(选定“范围”子句时)。

(2) WHILE<条件表达式>: 从当前记录开始(不选择“范围”短语,默认范围是 REST 时),对满足条件的记录进行操作,一旦遇到不满足条件的记录就停止,若选择范围,则从范围内第一条记录开始检查是否满足条件,遇到不满足条件的记录就停止。

## 1.2.3 VFP 项目管理器

项目是一种文件,用于跟踪创建应用系统所需要的所有文件。项目文件的扩展名是.pjx,项目说明文件的扩展名是.pjt。

项目管理器是开发 VFP 应用系统的一个工具,在 VFP 应用程序系统开发的过程中,各种格式的文件通过项目管理器捆绑在一起,开发人员通过项目管理器可以方便地建立应用程序的各个部分,集成管理所有与开发应用系统有关的各种类型的文件,免去设计者在应用系统开发过程中繁杂的管理工作。

### 1. 项目管理器对话框的组成和操作

项目管理器对话框主要由选项卡和命令按钮组成,如图 1-5 所示。

#### 1) 选项卡

(1) “全部”选项卡: 显示和管理项目中所有类型的文件。

(2) “数据”选项卡: 显示和管理项目中的所有数据,如数据库、数据表、查询和视图。

(3) “文档”选项卡: 显示和管理项目中的所有文档,如表单、报表和标签。

(4) “类”选项卡: 显示和管理由类设计器建立的类库文件。

(5) “代码”选项卡: 显示和管理程序文件、API 库文件和应用程序等。

(6) “其他”选项卡: 显示和管理菜单文件、文本文件,以及由 OLE 等工具建立的其他文件。

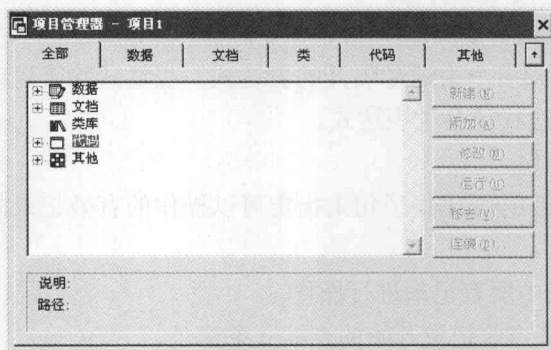


图 1-5 项目管理器对话框

## 2) 命令按钮

命令按钮用于对选定的文件进行操作，选定的文件不同，所进行的操作也就不同。常见的命令按钮如下。

- (1) “新建”按钮：用于建立新的文件或对象，类型为当前所选定的类型。
- (2) “添加”按钮：用于把已有的文件添加到项目中。
- (3) “修改”按钮：用于打开选定的文件或对象，并在相应地设计器中进行修改。
- (4) “运行”按钮：用于运行选定的查询、程序及表单等。
- (5) “浏览”按钮：用于在“浏览”窗口中打开一个数据表，以便查看其内容。
- (6) “移去”按钮：把选定的文件或对象从项目中移去或从磁盘上删除。
- (7) “连编”按钮：重新构造一个项目或应用程序，还可重新构造一个可执行文件。

## 3) 对话框的操作

在项目管理器对话框中，各个组成部分都是以图形的方式来组织和管理的，可以根据需要对话框位置、工具栏位置等进行设置。

(1) 移动和缩放：拖动对话框标题可移动窗口位置，拖拽对话框边框可调整对话框大小。

(2) 设置对话框位置：拖动项目管理器标题栏到系统工具栏下方，可以将项目管理器窗口放到工具栏上，若想还原，右击选项卡空白处，选择快捷菜单中的“拖走”命令即可。

(3) 设置折叠模式：单击项目管理器对话框右上方的折叠按钮(向上箭头)，此对话框将折叠，向上箭头变成向下箭头，若要还原单击向下箭头即可。

(4) 拆分项目管理器：当项目管理器是折叠方式或位于工具栏时，可用鼠标拖拽任一选项卡让它离开项目管理器。

## 2. 项目的建立

建立一个新项目可以采用菜单方式、项目向导和项目创建命令三种途径实现。

### 1) 利用菜单方式建立新项目

(1) 从“文件”菜单中选择“新建”命令，启动“新建”对话框，或单击工具栏中的“新建”图标，系统将弹出“新建”对话框，如图 1-6 所示。

