





# Visual Foxpro

## 编程方法与技巧

| 毕文才 编著 |

 郑州大学出版社


- ◎ 策划编辑 何晓红
- ◎ 责任编辑 吴 昊
- ◎ 责任校对 张锦森
- ◎ 封面设计 
- ◎ 版式设计 小羽毛

ISBN 978-7-81106-986-0



9 787811 069860 >


定价:38.00元



# Visual Foxpro

## 编程方法与技巧

| 毕文才 编著 |

 郑州大学出版社

**图书在版编目(CIP)数据**

Visual Foxpro 编程方法与技巧/毕文才编著. —郑州:  
郑州大学出版社, 2008. 12

ISBN 978 - 7 - 81106 - 986 - 0

I. V… II. 毕… III. 关系数据库 - 数据库管理系统,  
Visual Foxpro - 程序设计 IV. TP311. 138

中国版本图书馆 CIP 数据核字 (2008) 第 178923 号

郑州大学出版社出版发行

郑州市大学路 40 号

出版人: 邓世平

全国新华书店经销

河南省瑞光印务股份有限公司印制

开本: 787 mm × 1 092 mm

印张: 12

字数: 341 千字

版次: 2008 年 12 月第 1 版

邮政编码: 450052

发行部电话: 0371 - 66966070

1/16

印次: 2008 年 12 月第 1 次印刷

---

书号: ISBN 978 - 7 - 81106 - 986 - 0

定价: 38.00 元

本书如有印装质量问题, 由本社负责调换

## 内容提要

## NEIRONGTIYAO

本书针对 Visual Foxpro 程序编写中常见的问题,进行了深入透彻的分析。从实际出发,既有理论的阐述又有应用的例子,侧重于编程的方法与技巧的描述。

全书分为程序设计、界面设计、Activex 控件、数据处理、报表设计、API 函数的应用六部分,适用于具有一定 Visual Foxpro 基础知识的读者阅读。

## 前言

## QIANYAN

Visual Foxpro 是 Microsoft 公司推出的面向对象可视化编程的数据库软件。经历了由 dBASE、FoxBASE 到 Visual Foxpro 的发展历程。目前, Visual Foxpro 最高版本为 Visual Foxpro 9.0。版本的不断提升,充分说明了 Visual Foxpro 有着旺盛的生命力和广阔的发展前景,深受广大编程爱好者和专业人员的厚爱。

Visual Foxpro 在中小型数据应用系统的开发中占有很大的市场份额,以其数据功能强、可视化程度高、简单易学等特点著称。但对于初学 Visual Foxpro 的读者来讲,要想具备一定的程序编写能力也不是一件太容易的事情。

为此,作者从 Visual Foxpro 程序设计的实际出发,结合多年来从事 Visual Foxpro 编程实践的经验,深入浅出、讲究实用,围绕编程的方法与技巧分程序设计、界面设计、ActiveX 控件、数据处理、报表设计、API 函数的应用六部分进行了详细的阐述。在成书时还在附录中加了常用命令、函数、属性、方法、事件的介绍,以便读者查阅。

本书所有的编程应用的例子和代码均在 Visual Foxpro 中进行了验证,插图中程序运行显示的界面,都是程序在 Visual Foxpro 验证通过时在计算机上捕获的。注重方法技巧,突出实际应用是编写此书的宗旨。在编写的过程中查阅了大量的资料,无论是理论阐述,还是应用举例,都力争做到严谨、详实、清晰、可靠,以增加本书的可读性。

由于水平有限,时间仓促难免有一些错误和不足之处,敬请广大读者批评指正。

# 目录

## MULU

<b>第 1 章 程序设计</b> .....	1
1.1 主程序与退出程序 .....	1
1.1.1 什么是主程序 .....	1
1.1.2 编写主程序 .....	2
1.1.3 退出程序 .....	3
1.2 自定义函数与过程 .....	4
1.2.1 自定义函数 .....	5
1.2.2 过程 .....	6
1.3 项目管理 .....	8
1.3.1 建立项目 .....	9
1.3.2 项目管理器 .....	11
1.4 密码处理 .....	12
1.4.1 改变 dbf 表的后缀名加密 .....	12
1.4.2 运用求余算法加密 .....	13
1.5 汉字的区位码 .....	16
1.6 用户权限的控制 .....	17
1.7 获取网卡号 .....	19
1.7.1 获取本机 IP 地址 .....	19
1.7.2 获取网卡号 .....	20
1.8 客户/服务器(C/S)程序的设计 .....	21
1.8.1 VFP 程序和数据库的安装 .....	21
1.8.2 数据库的访问与连接 .....	21
<b>第 2 章 用户界面设计</b> .....	25
2.1 如何关闭菜单中的工具栏 .....	25
2.2 如何自动改变表单的背景颜色 .....	27

2.3	系统登录界面的制作 .....	28
2.3.1	为 Visual Foxpro 主窗口加入背景图像 .....	28
2.3.2	制作登录表单 .....	29
2.4	滚动字幕的制作 .....	31
2.5	倒计时器的制作 .....	32
2.6	隐藏 Visual Foxpro 的主界面 .....	35
2.7	制作圆形窗口 .....	36
2.8	字符的淡出淡入 .....	38
2.9	系统信息表单 .....	41
2.10	播放 Flash 动画 .....	42
2.11	定制工具栏 .....	45
2.12	彩色按钮的设计 .....	48
2.13	添加背景音乐 .....	49
<b>第3章</b>	<b>Activex 控件 .....</b>	<b>51</b>
3.1	Activex 控件注册 .....	51
3.1.1	REGSVR32.EXE 手动注册 .....	51
3.1.2	Visual Foxpro 注册 .....	52
3.1.3	安装程序注册 .....	53
3.1.4	API 函数注册 .....	53
3.2	进度条控件_progressbar .....	54
3.2.1	添加进度条控件 .....	54
3.2.2	正确使用进度条控件 .....	54
3.3	richtextbox 控件 .....	56
3.3.1	主要属性 .....	56
3.3.2	主要方法 .....	57
3.3.3	应用举例 .....	57
3.4	Kodak 扫描控件与编辑控件 .....	58
3.4.1	Kodak 扫描控件 .....	59
3.4.2	kodak 编辑控件 .....	61
3.4.3	应用举例 .....	62
3.5	TreeView 控件 .....	64
3.5.1	主要的属性方法事件 .....	64
3.5.2	应用举例 .....	66
3.6	ImageList 控件 .....	70
3.6.1	主要的属性和方法 .....	70
3.6.2	应用举例 .....	71
3.7	ProgressBar 控件 .....	71
3.7.1	主要属性与方法 .....	72



3.7.2 应用举例 .....	72
3.8 Mschart 控件 .....	73
3.8.1 主要的属性 .....	73
3.8.2 主要的方法与事件 .....	75
3.8.3 应用举例 .....	76
<b>第4章 数据处理</b> .....	<b>80</b>
4.1 序列号的生成 .....	80
4.2 随机数据的获取 .....	81
4.2.1 rand()函数 .....	82
4.2.2 应用举例 .....	82
4.3 求两个正整数的最大公约数和最小公倍数 .....	84
4.4 使用 GetFile()函数与宏替换打开表 .....	86
4.4.1 GetFile()函数 .....	86
4.4.2 应用举例 .....	88
4.5 自动产生所输入汉字的拼音 .....	88
4.6 运用 select 语句进行查询统计 .....	91
4.6.1 select 语句 .....	91
4.6.2 运用 select 语句进行查询 .....	93
4.6.3 运用 select 语句进行统计 .....	101
4.7 如何将金额由小写变为大写 .....	114
4.7.1 定义函数 .....	114
4.7.2 调用函数 .....	116
4.8 运用 ALTER TABLE 命令实现统计功能 .....	117
4.8.1 ALTER TABLE 命令 .....	117
4.8.2 应用举例 .....	117
4.9 将 DBF 表导出到 WORD .....	118
<b>第5章 报表设计</b> .....	<b>122</b>
5.1 报表中各个带区的作用 .....	122
5.2 控制报表每页打印的记录数 .....	123
5.3 图形统计报表 .....	126
5.3.1 将统计图形加入 DBF 的通用字段 .....	127
5.3.2 将统计图形加入报表 .....	130
5.4 调用 Excel 打印报表 .....	131
5.5 带有照片的报表 .....	133
5.5.1 图片/Activex 绑定控件(图文框) .....	134
5.5.2 应用举例 .....	135
5.6 报表的打印 .....	136
5.6.1 如何打印指定的页 .....	136

5.6.2	如何计算总页数,并每页打印“第 x 页 共 y 页”字样 .....	137
5.6.3	如何使计算机打印纸张类型适合于所设计报表 .....	137
5.6.4	数据库当前记录的打印 .....	139
5.7	收费票据报表 .....	140
5.8	.FRX 文件的分析 .....	142
<b>第 6 章</b>	<b>API 函数的应用 .....</b>	<b>147</b>
6.1	API 函数的声明 .....	147
6.2	API 函数打开文件 .....	149
6.3	建立网络连接 .....	151
6.3.1	WNetAddConnection2() 函数 .....	151
6.3.1	函数的应用 .....	152
6.4	启动邮件程序 .....	154
6.5	防止同一程序被重复执行 .....	156
6.6	异形表单的制作 .....	159
6.6.1	函数 CreateRoundRectRgn() .....	159
6.6.2	函数 CreateEllipticRgn() .....	160
6.6.3	函数 CreatePolygonRgn() .....	161
6.6.4	函数 CreateRectRgn() .....	162
6.7	制作多媒体徽标 .....	162
6.7.1	函数介绍 .....	163
6.7.2	应用举例 .....	164
<b>附 录</b>	.....	<b>166</b>
常用 VFP 命令、函数及程序语句	.....	166
一、常用命令	.....	166
二、常用函数	.....	171
三、主要程序语句	.....	174
常用属性、事件及方法	.....	177

# 第 1 章 程序设计

## 1.1 主程序与退出程序

主程序是在程序编译时和编译后执行时最先调用的文件,是整个程序的入口。退出程序是程序结束时调用的文件,是整个程序的出口。这两个程序把握着所开发程序的进出口,是进行程序设计时必须解决的问题。

### 1.1.1 什么是主程序

主程序是应用程序中第一个执行的程序,决定着整个程序执行的顺序。主要包括以下内容。

配置执行环境。就是要为应用程序配置一个在执行时必需的运行环境,以保证程序的正常运行,如数据库的打开方式,菜单的显示方式等。

确定默认路径和目录。为程序在执行过程中调用文件提供可搜索的路径和目录,是程序设计时表单、菜单、报表、图片、.prg 等文件分类保存的路径和目录。

设置退出程序。是程序退出时必须执行的文件,也就是调用命令 quit 时执行的文件(详细见 1.1.3)。

读取程序的起始点。程序开始运行的入口,登录应用程序所需的用户名密码的录入要在这里完成。

调用初始界面。登录应用程序执行完成后计算机上显示的程序的主界面,程序的所有操作都要在该界面下完成。

主程序文件可以是一个表单、菜单或 .prg 文件,要在的项目文件中将主程序文件设置为主文件,具体如图 1-1。

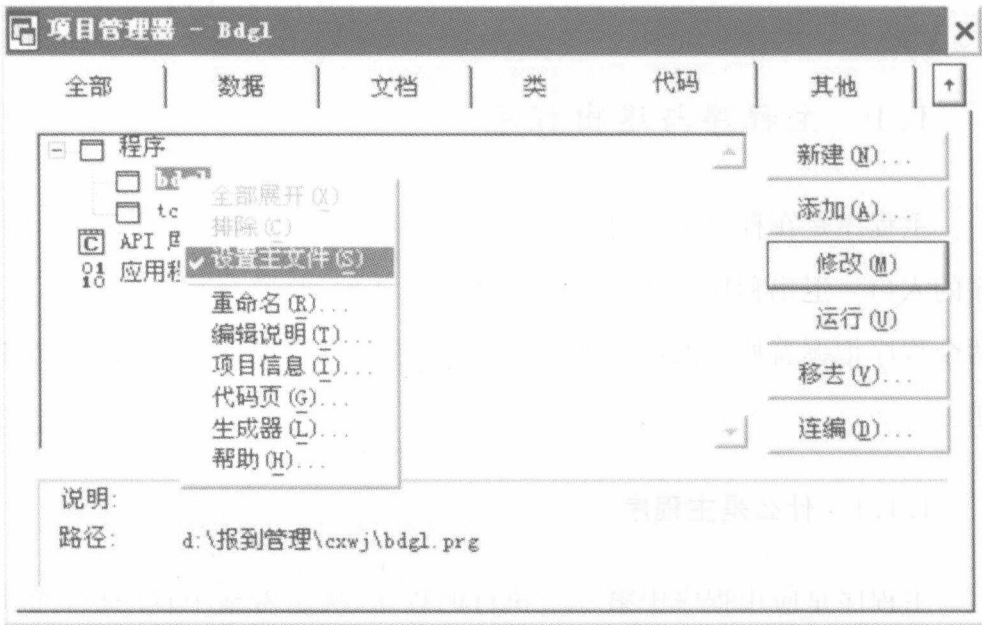


图 1-1 设置主程序

### 1.1.2 编写主程序

主程序一般是一个 .prg 文件,下边以一个简单的主程序代码为例来进行说明。

&& 配置执行环境

set sysmenu off && 禁止菜单栏显示

set exclusive off && 网络用户均可以打开表

set talk off && 菜单的信息栏不显示数据

public ljm, syqx, qx11, qx12, qx13;

qx21, qx22, qx23;

qx31, qx32, qx33, qx34;

qx41, qx42, qx43 && 定义程序执行时所用的全局变量

.....

&& 确定默认路径和目录

set path to . \, . \data, . \picture

&& 设置退出程序

on shutdown do tc. prg &&tc. prg 是一个退出程序

&& 读取程序的起始点

Do form dl

&& 调用初始界面

do sxgl. mpr

Read events && 开始处理事务

Read events 命令是主程序中必不可少的一条命令,主要功能是通知操作系统开始进行 Visual Foxpro 事务处理,如果缺少该命令,执行时 Visual Foxpro 的主界面只会一扫而过,而不会运行应用程序,并且在关机时还会出现不能退出 Visual Foxpro 的情况。

### 1.1.3 退出程序

退出程序是应用程序退出时执行的一个 . prg 文件,由 on shutdown 命令在主程序中来指定执行的一个文件。主程序指定后每次退出应用程序

或调用 quit 命令都要执行这个文件。下面给出一个最简单的退出程序 te. prg 。

```
xy = messagebox(“退出请点确定”,1 + 64 + 0,“信息”)
if xy = 1
clear events && 结束事务处理。
quit
endif
```

单击主窗口的关闭按钮或执行 quit 命令出现图 1 - 2 中的界面。

clear events 命令是退出程序中必不可少的一条命令,否则,就会在单击主窗口的关闭按钮或执行 quit 命令时出现不能退出 Visual Foxpro 的提示。

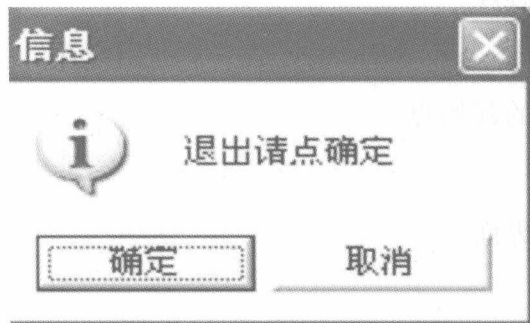


图 1 - 2 程序退出时的运行界面

## 1.2 自定义函数与过程

为了提高设计效率,在程序设计中不重复进行代码的编写,Visual Foxpro 有自定义函数与过程的功能。二者非常类似,都是以 . prg 文件格式编写和保存,调用和输出结果存在差异。

### 1.2.1 自定义函数

#### (1) 创建自定义函数

可以用 FUNCTION 命令来定义函数,语法格式如下:

```
FUNCTION functionName
```

```
    [ LPARAMETERS parameter1 [ ,parameter2... ] ]
```

```
    Commands
```

```
    [ RETURN[ eExpression ] ]
```

```
ENDFUNC
```

- 或者 -

```
FUNCTION functionName( [ parameter1 [ ,parameter2 ,... ] ] )
```

```
    Commands
```

```
    [ RETURN[ eExpression ] ]
```

```
ENDFUNC
```

FunctionName 用户自定义的函数名,最长不超过 245 个字符。

LPARAMETERS parameter1 是定义接收参数。

Commands 完成函数操作的命令集。

eExpression 自定义函数的输出值。

下面以自定义函数来求两个数的和,文件名为 qsun. prg

```
FUNCTION nsun
```

```
    LPARAMETERS x1 ,x2
```

```
    x3 = x1 + x2
```

```
    RETURN x3
```

```
ENDFUNC
```

上述代码可以 . prg 文件格式单独保存,使用时再调用,也可以保存在运行自定义函数时已运行的程序文件中。

## (2) 调用自定义函数

1) 调用单独保存的函数, 例如:

```
SET PROCEDURE TO qsun. prg
```

```
? nsun(20,40)
```

```
SET PROCEDURE TO
```

2) 调用已运行的程序中的函数, 例如:

```
.....
```

```
? nsun(20,40)
```

```
FUNCTION nsun
```

```
    LPARAMETERS x1, x2
```

```
    x3 = x1 + x2
```

```
    RETURN x3
```

```
ENDFUNC
```

```
.....
```

### 1.2.2 过程

过程的定义同函数的定义相似, 要先进行定义再调用。

#### (1) 创建过程

用 PROCEDURE 命令创建, 其语法格式如下:

```
PROCEDURE PName
```

```
    [ LPARAMETERS parameter1 [ , parameter2 , ... ] ]
```

```
    Commands
```

```
    [ RETURN [ eExpression ] ]
```

```
ENDPROC
```

- 或者 -

```
PROCEDURE ProgramName( [ parameter1 [ , parameter2 , ... ] ] )
```



### Commands

```
[ RETURN[ eExpression ] ]
```

ENDPROC

ProgramName 是要建立的过程名,最长 254 个字符。

LPARAMETERS parameter1 是定义接收参数。

Commands 完成过程操作的命令集。

eExpression 过程的输出值。

下面一个过程来求两个数的和,文件名为 qsun. prg

```
PROCEDURE nsun
```

```
LPARAMETERS x1 ,x2
```

```
    x3 = x1 + x2
```

```
    RETURN x3
```

```
ENDPROC
```

上述代码可以 . prg 文件格式单独保存,使用时再调用,也可以保存在运行自定义过程时已运行的程序文件中。

### (2) 过程的调用

1) 调用单独保存的过程,

使用 DO 命令进行过程调用,语法格式如下:

```
DO ProgramName [ IN ParameterName ] [ WITH[ ParameterList ] ]
```

ProgramName 过程名,ParameterName 过程所文件的文件名,ParameterList 要传入的参数,最多为 27 个。

```
DO nsun IN qsun. prg WITH 20 ,40
```

```
或? nsun(20 ,40)
```

2) 调用已运行的程序中的过程,例如:

```
.....
```

```
? nsun(20 ,40)
```