



並通京筆教育“十三五”规划教材

Visual FoxPro

数据库与程序设计实验教程

杨 永 周 凯 吴明涛 主编
杨王黎 主审



中国石化出版社

[HTTP://WWW.SINOPEC-PRESS.COM](http://www.sinopec-press.com)

普通高等教育“十三五”规划教材

Visual FoxPro 数据库与 程序设计实验教程

杨 永 周 凯 吴明涛 主 编
杨王黎 主 审



中国石化出版社

内 容 提 要

本书是与《Visual FoxPro 数据库与程序设计》(中国石化出版社,杨永、周凯、吴明涛主编)教材配套的实验教材。全书共分四个部分:第一部分是 Visual FoxPro 数据库与程序设计实验指导,安排了 21 个实验,每个实验给出了实验目的、相关知识点及实验内容,这部分的实验不提供参考答案,以锻炼学生的独立学习能力;第二部分是习题集,通过选择、判断、程序设计等题型让学生掌握 Visual FoxPro 的基本语法和程序设计能力;第三部分是习题解答,提供第二部分全部习题的参考答案;第四部分是计算机综合应用实验指导,以开发“学生成绩管理系统”项目为例,给出开发的全过程,帮助提高学生的综合应用能力。

本书适合作为高等学校非计算机专业本科生的程序设计实验教材,实验学时建议为 30~40 学时;本书也可以作为软件技术人员的学习参考书。

图书在版编目(CIP)数据

Visual FoxPro 数据库与程序设计实验教程 / 杨永, 周凯, 吴明涛主编. —北京:中国石化出版社, 2018. 1
普通高等教育“十三五”规划教材
ISBN 978-7-5114-4797-5

I. ①V… II. ①杨… ②周… ③吴… 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(2017)第 328366 号

未经本社书面授权,本书任何部分不得被复制、抄袭,或者以任何形式或任何方式传播。版权所有,侵权必究。

中国石化出版社出版发行

地址:北京市朝阳区吉市口路 9 号

邮编:100020 电话:(010)59964500

发行部电话:(010)59964526

<http://www.sinopec-press.com>

E-mail:press@sinopec.com

北京科信印刷有限公司印刷

全国各地新华书店经销

*

787×1092 毫米 16 开本 12.5 印张 312 千字

2018 年 2 月第 1 版 2018 年 2 月第 1 次印刷

定价:30.00 元

前 言

本书是与杨永、周凯、吴明涛等编写的《Visual FoxPro 数据库与程序设计》教材相配套的实验指导书和习题集，是按照知识点的循序渐进思路进行编写，其中包含了作者以及同事十几年从事 Visual FoxPro 程序设计教学的经验和总结。

根据中国高等学校计算机基础教育课程体系(CFC 2008)的精神，本着提高学生计算机应用能力和注重实践教学的宗旨，本书第一部分给出了 21 个实验，这些实验既有验证性实验，也有设计性实验，还有综合性实验。每个实验给出了实验的目的，相关的知识点及实验内容。学生通过上机完成这些实验，可以很好地锻炼自己的程序设计能力，提高计算机的应用水平。第二部分是每章的习题，作者根据多年的教学经验，编写了选择题、判断题、程序设计和窗体设计等多种题型，力求全方位促进相关知识的掌握。第三部分是习题解答，提供第二部分全部内容的参考答案。第四部分计算机综合应用实验指导，以开发“学生成绩管理系统”项目为例，给出开发的全过程，用以提高学生的综合应用能力。

本书涉及的实验和习题较多，也比较全面，各位教师可以根据自己学校的授课情况选择其中的部分实验内容，同时根据具体的教学进度，调整实验次数。书中内容的设计能够满足不同层次学生的不同学习需要。读者在应用本书实验指导部分的时候一定要多练习、多实践，要在“做中学”；而在应用习题部分一定要多分析，不要急于看答案，以便更好地帮助学生提高程序设计能力。

本书是在多年 Visual FoxPro 程序设计教学的基础上，以提高学生综合编程能力为出发点编写的，本书由东北石油大学杨永、周凯和吴明涛主编，杨永编写了第一部分 Visual FoxPro 数据库与程序设计实验指导，吴明涛编写了第二部分 Visual FoxPro 程序设计习题集和第三部分 Visual FoxPro 程序设计习题解答，周凯编写了第四部分计算机综合应用实验指导，杨永统稿全书。在本书编写及以往应用过程中，东北石油大学计算机基础教育系的教师们提出了许多宝贵的

意见，在此表示衷心的感谢，同时也参考了一些专家、学者的真知灼见和网上资源，在此深表谢意。

由于编者水平有限，书中难免有一些疏漏之处，恳请广大读者提出宝贵意见，我们将衷心感谢！

编 者
2018 年 1 月

目 录

第一部分 Visual FoxPro 数据库与程序设计实验指导

实验 1	Visual FoxPro 的基础知识	(2)
实验 2	顺序结构和选择结构	(9)
实验 3	一重循环程序设计—DO WHILE 循环	(16)
实验 4	一重循环程序设计—FOR 循环	(20)
实验 5	多重循环程序设计	(23)
实验 6	数 组	(27)
实验 7	模块化程序设计	(32)
实验 8	表的建立与维护	(38)
实验 9	表的排序、索引与统计	(41)
实验 10	多表操作	(45)
实验 11	数据库的操作	(47)
实验 12	和表相关的程序设计	(49)
实验 13	查询与视图设计	(53)
实验 14	SQL 语言的应用	(55)
实验 15	项目管理器使用	(57)
实验 16	面向对象程序设计	(59)
实验 17	表单设计(一)	(63)
实验 18	表单设计(二)	(66)
实验 19	表单设计(三)	(75)
实验 20	学生成绩管理系统	(83)
实验 21	报表和菜单设计	(85)

第二部分 Visual FoxPro 程序设计习题集

第 1 章	数据库系统基础知识	(88)
第 2 章	Visual FoxPro 的数据及其运算	(90)
第 3 章	结构化程序设计	(95)
第 4 章	表的基本操作	(109)
第 5 章	数据库的基本操作	(114)
第 6 章	表单设计	(115)

第 7 章 SQL 语言的应用 (125)

第 8 章 查询与视图设计 (129)

第 9 章 菜单与报表设计 (130)

第三部分 Visual FoxPro 程序设计习题解答

第 1 章 数据库系统基础知识 (132)

第 2 章 Visual FoxPro 的数据及其运算 (132)

第 3 章 结构化程序设计 (133)

第 4 章 表的基本操作 (136)

第 5 章 数据库的基本操作 (136)

第 6 章 表单设计 (137)

第 7 章 SQL 语言的应用 (137)

第 8 章 查询与视图设计 (138)

第 9 章 菜单与报表设计 (138)

第四部分 计算机综合应用实验指导

实验 1 建立“学生成绩管理系统”项目、数据库和表 (140)

实验 2 登录界面及系统功能的设计与实现 (147)

实验 3 菜单的设计与实现 (155)

实验 4 “学生成绩管理系统”数据录入功能的设计与实现 (160)

实验 5 “学生成绩管理系统”的数据修改 (168)

实验 6 “学生成绩管理系统”中的数据查询 (179)

实验 7 报表及主程序文件 (188)

第一部分»

Visual FoxPro数据库与 程序设计实验指导

实验 1 Visual FoxPro 的基础知识

一、实验目的

1. 掌握 VFP 6.0 的启动、退出及窗口组成;
2. 熟悉 VFP 程序的运行环境和 VFP 常用命令子句;
3. 掌握 VFP 的表达式和各种函数的使用;
4. 掌握 VFP 常量和变量的使用。

二、实验内容

(一) VFP 6.0 的启动和退出

1. 启动

- (1) 双击桌面上 VFP 的快捷方式;
- (2) 单击“开始/程序”项,在程序菜单中选择“Microsoft Visual FoxPro 6.0”下的“Microsoft Visual FoxPro 6.0”菜单项就可以启动 VFP 6.0。

2. 退出

- (1) 单击标题栏的关闭按钮;
- (2) 双击 VFP 窗口标题栏左边的控制按钮;
- (3) 单击“文件”菜单中的“退出”;
- (4) 在“命令窗口”中输入命令“QUIT”。

(二) 表达式和函数的使用

掌握以下 VFP6.0 常用的函数:

1. 数值函数

- 绝对值函数 ABS(<数值表达式>);
平方根函数 SQRT(<数值表达式>);
指数函数 EXP(<数值表达式>);
取整函数 INT(<数值表达式>);
求模(余数)函数 MOD(<数值表达式 1>, <数值表达式 2>);
四舍五入函数 ROUND(<数值表达式 1>, <数值表达式 2>);
求圆周率 π 的近似值函数 PI();
最大值函数 MAX(<表达式清单>);
最小值函数 MIN(<表达式清单>)。

2. 字符函数

- 求字符串长度函数 LEN(<字符串表达式>);

查找子串位置函数 AT(<子字符串>, <主字符串>, [数值表达式]);
左截取子串函数 LEFT(<字符表达式>, <数值表达式>);
右截取子串函数 RIGHT(<字符表达式>, <数值表达式>);
取子串函数 SUBSTR(<字符表达式>, <数值表达式 1>[, <数值表达式 2>]);
删除字符串尾部空格函数 RTRIM(<字符表达式>)或 TRIM(<字符表达式>);
删除字符串前导空格函数 LTRIM(<字符表达式>);
删除字符串前导和尾部空格函数 ALLTRIM(<字符表达式>);
空格生成函数 SPACE(<数值表达式>);
小写变大写函数 UPPER(<字符表达式>);
大写变小写函数 LOWER(<字符表达式>);
宏替换函数 &<字符型内存变量>。

3. 日期与时间函数

系统日期函数 DATE();
系统时间函数 TIME();
日期函数 DAY(<日期表达式>);
月份函数 MONTH(<日期表达式>);
星期几函数 DOW(<日期表达式>)或 CDOW(<日期表达式>);
年份函数 YEAR(<日期表达式>)。

4. 转换函数

求字符 ASCII 码函数 ASC(<字符串表达式>);
将 ASCII 码转换相应字符函数 CHR(<数值表达式>);
字符型转换成日期型函数 CTOD(<字符表达式>);
日期型转换成字符型函数 DTOC(<日期表达式>);
数值型转换成字符型函数 STR(<数值表达式>, [<长度>[, <小数位数>]]);
将字符串转换成数值函数 VAL(<字符表达式>)。

5. 测试函数

数据类型测试函数 TYPE(<表达式>)或 VARTYPE(<表达式>);
表头测试函数 BOF();
表尾测试函数 EOF();
记录号测试函数 RECNO();
记录个数测试函数 RECCOUNT();
查找是否成功测试函数 FOUND();
文件存在测试函数 FILE();
条件函数 IIF(<逻辑型表达式>, <表达式 1>, <表达式 2>);
判断值介于两个值之间的函数 BETWEEN(<被测试表达式>, <下限表达式>, <上限表达式>)。

(三) 变量

1. 内存变量: 内存变量是一种独立于数据库而与内存有关的变量, 它用以存放数据处理过程中的常量、中间结果或最终结果。

2. 内存变量的命名: 由 1—128 个字母、数字、汉字、下划线组成的以字母、下划线或汉字开头的字符序列。

3. 内存变量的赋值(1) <内存变量名>=<表达式>

(2) STORE <表达式> TO <内存变量名表>

4. 内存变量的显示:

LIST | DISPLAY MEMORY [LIKE<内存变量名>][TO PRINTER] | [TO FILE<文件名>]

(四) 实验习题

1. 分别给下列内存变量赋值。

(1) 用“=”命令赋值

MA="*****"

MB=123

MC=.T.

MD={^2009/03/01}

? MA, MB, MC, MD

(2) 用“STORE”命令将 NA, NB, NC 赋值为 0, 将 PA、PB、PC 分别赋值为-8、“计算机”和.F.。

STORE 0 TO NA, NB, NC

? NA, NB, NC

STORE -8 TO PA

STORE "计算机" TO PB

STORE .F. TO PC

? PA, PB, PC

2. 函数使用

根据要求写出函数命令, 并记录显示结果。

(1) 计算"Visual FoxPro 6.0"的长度;

(2) 利用字符函数在字符串"东北石油"和"大学"之间插入 3 个空格;

(3) 分别求字符串"东北石油大学"的子字符串"东北"、"石油"和"大学";

(4) 将字符串"Visual FoxPro 6.0"中的字母转换为大写字母;

(5) 将字符串"Visual FoxPro 6.0"中的字母转换为小写字母;

(6) 将字符串"Visual FoxPro 6.0"中尾部的空格去掉;

(7) 将字符串"Visual FoxPro 6.0"中前导空格去掉;

(8) 将字符串"Visual FoxPro 6.0"中前导和尾部的空格去掉;

- (9) 求字符串 "is" 在字符串 "This is a book." 中第一次和第二次出现的起始位置;
- (10) 对 3.1415926 进行四舍五入保留 3 位小数;
- (11) 对 3.1415926 进行取整运算;
- (12) 分别求 10 除以 3, 10 除以 -3, -10 除以 3 和 -10 除以 -3 的余数;
- (13) 已知 N="125.6", 利用 N 求数据 125.6 与 98.9 的和;
- (14) 已知 Var1=5, Var2=9, 测试 Var1 是否在 1 到 10 之间, 并用表达式判断 Var2 是否小于 Var1?
- (15) 分别求当前系统的日期和时间, 并测试其数据类型;
- (16) 分别求当前系统的年、月、日和星期几;
- (17) 把数值 21.265389 转换为字符串, 宽度 11, 保留 4 位小数;
- (18) 将 "3.14ab" 转换为数值型数据, 并测试其结果类型;
- (19) 将字符串 "03/01/2013" 转换为日期型数据, 并测试其结果类型;
- (20) 设 $x=3.5$, $y=2$, 求表达式 $\frac{x^3+y^4}{\sqrt{x+y}-x/y}$ 的值。

3. 写出下面表达式的结果。

- (1) INT(ABS(-5.7))
- (2) STR(16.35749, 7, 3)
- (3) SUBSTR("东北石油大学", 5, 4)
- (4) ROUND(15.81, 1) < INT(15.81)
- (5) INT(6.26 * 2) % ROUND(3.14, 0) 、

4. 对话框函数 MESSAGEBOX()

MESSAGEBOX() 函数可以显示一个自定义的对话框。常用作提示之用, 也可以作一些简单的选择, 比如“确定”、“取消”等, 是程序中经常用到的一个函数。

函数形式:

MESSAGEBOX(提示信息[, 对话框的类型[, 对话框窗口标题]])

返回值类型:

数值型

参数描述:

提示信息：表示对话框中所显示的提示文字，为字符型。

对话框的类型：用于确定对话框的按钮、图标等属性，这是一个数值型的参数。

对话框的类型设置：

对话框的类型设置包括设置按钮、图标和设置默认按钮 3 部分的内容，其中可以单独设置一种或几种的组合。

(1) 设置按钮属性

按钮属性设置如表 1-1 所示。

表 1-1 MESSAGEBOX 函数按钮属性设置

值	对话框按钮属性
0	仅有一个“确定”按钮
1	有“确定”和“取消”按钮
2	有三个按钮，分别是“终止”、“重试”、“忽略”
3	“是”、“否”和“取消”按钮
4	“是”和“否”
5	“重试”和“取消”

比如在命令窗口中输入：MESSAGEBOX('是否真的要退出系统?', 4)后回车，则弹出如图 1-1 所示对话框：

若第 2 个参数省略则默认值是 0，此时将弹出如图 1-2 所示对话框：



图 1-1 含有两个按钮的 MESSAGEBOX 示例



图 1-2 含有一个按钮的 MESSAGEBOX 示例

(2) 设置图标

图标设置如表 1-2 所示。

表 1-2 MESSAGEBOX 函数图标属性设置

值	图 标	值	图 标
16	红色叉号	48	感叹号
32	问号	64	字母 i

比如在命令窗口中输入：MESSAGEBOX('是否真的要退出系统?', 48)后回车，则弹出如图 1-3 所示对话框。



图 1-3 MESSAGEBOX 函数含有图标示例



图 1-4 MESSAGEBOX 函数含有图标与按钮示例

如果又想要问号图标，又想要“是”、“否”两个按钮，可以按照如下的方法完成，输入 MESSAGEBOX('是否真的要退出系统?', 4+32)，则弹出如图 1-4 所示对话框。

(3) 设置默认按钮

所谓设置默认按钮就是在有按钮的情况下，按下回车键执行的那个按钮，其值设置如表 1-3 所示。

表 1-3 MESSAGEBOX 函数默认按钮属性设置

值	默认按钮	值	默认按钮
0	第一个按钮	512	第三个按钮
256	第二个按钮		

例如：MESSAGEBOX('是否真的要退出系统?', 4+32)，这一句，您希望显示对话框时，默认的按钮为“否”，也就是按下“回车”键即执行“否”，那么就写成如下形式：

MESSAGEBOX('是否真的要退出系统?', 4+32+256)，则弹出如图 1-5 所示对话框。会发现在此时“否”按钮则默认为选中状态，若按回车键则执行该按钮。

① 对话框窗口标题：

用来显示在对话框窗口上部，那个蓝色区域内的信息。例如：

MESSAGEBOX('是否真的要退出系统?', 4+32+256, '注意')，则弹出如图 1-6 所示对话框。



图 1-5 MESSAGEBOX 函数具有默认按钮示例

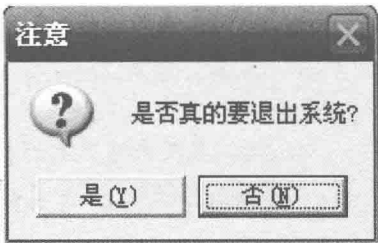


图 1-6 MESSAGEBOX 函数完整示例

如果所设的数大于按钮的数，比如设为 512，但只有两个按钮，则默认是第一个按钮，若缺省则默认显示标题为“Microsoft Visual FoxPro”。

② 函数返回值：

说明在对话框中按了不同的键，该函数将返回不同的值，键值对应如表 1-4 所示。

表 1-4 MESSAGEBOX 函数按键对应的值

值	键	值	键
1	确定	5	忽略
2	取消	6	是
3	终止	7	否
4	重试		

例如在命令窗口输入？ MESSAGEBOX('是否真的要退出系统?', 3+32+256, '注意') 并回车，则弹出如图 1-7 所示对话框。

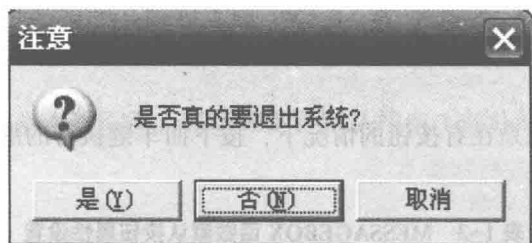


图 1-7 MESSAGEBOX 函数示例

此时，若用鼠标单击“是”按钮，则输出 6；单击“否”按钮，则输出 7；单击“取消”按钮，则输出 2。

实验 2 顺序结构和选择结构

一、实验目的

1. 掌握程序的建立和修改的基本方法；
2. 掌握程序设计中常用命令的使用方法；
3. 掌握数据输入和输出的命令；
4. 掌握 IF 和 DO CASE 两种选择结构的用法。

二、实验内容

(一) 要点复习

1. 程序文件的建立和运行

建立命令：MODIFY COMMAND <程序文件名>

运行命令：DO <程序文件名>

2. 程序设计中的常用命令

(1) 交互式输入语句

INPUT [<字符型表达式>] TO [<内存变量>]

ACCEPT [<字符型表达式>] TO [<内存变量>]

WAIT [<字符型表达式>] TO [<内存变量>]

(2) 格式化输入输出语句

@<行号, 列号> [SAY <表达式 1>] GET <变量名> [DEFAULT<表达式 2>]

READ

(3) 注释语句

* <注释行信息, 表示整行都为注释, 程序执行时不执行以 * 开头的语句>

&& <在语句的尾部加注释>

(4) 终止命令和退出命令

终止命令：CANCEL

退出命令：QUIT

(5) 返回命令

RETURN [<表达式>] | TO MASTER | TO <程序名>

3. 选择结构语句

(1) 单分支选择结构

语句格式：

IF<条件表达式>

<命令序列>

ENDIF

(2) 双分支选择结构

语句格式:

IF<条件表达式>

<命令序列 1>

ELSE

<命令序列 2>

ENDIF

(3) 分支结构的嵌套

对上述分支结构中的<命令序列>, 可以是包含任何 Visual FoxPro 命令语句, 也可以包括一个或几个合法的分支结构语句, 也就是说分支结构可以嵌套。对于嵌套的分支结构语句, 一定注意内外层分支结构层次分明, 即注意各个层次的 IF...ELSE...ENDIF 语句配对情况。

(4) 多分支选择结构

语句格式:

DOCASE

CASE <条件表达式 1>

<命令序列 1>

CASE <条件表达式 2>

<命令序列 2>

...

CASE <条件表达式 n>

<命令序列 n>

[OTHERWISE

<命令序列 n+1>]

ENDCASE

(二) 实验习题

1. 程序填空

(1) 已知三角形的三边, 求三角形面积。三角形的面积公式为: $AREA = \sqrt{S * (S - A) * (S - B) * (S - C)}$, 其中 $S = (A + B + C) / 2$, A、B、C 为三角形三条边的长。

SET TALK OFF

CLEAR

INPUT "A=" TO A

INPUT "B=" TO B

INPUT "C=" TO C

【1】 $A + B > C$ AND $A + C > B$ AND $B + C > A$

$S = (A + B + C) / 2$