



赠送光盘
中附有完
整的案例
源代码

Visual FoxPro 课程设计 案例精编

罗 颂 詹 斌 余 健 编著

- 考勤管理系统 • 客户管理系统 •
- 综合人事管理系统 • 订单管理系统 •
- 学位论文管理系统 • 超市进销存管理系统 •
- 学生成绩管理系统 • 程序的完善与发布 •



清华大学出版社

高等院校课程设计案例精编

Visual FoxPro 课程设计案例精编

罗 颂 詹 斌 余 健 编著

清华大学出版社

北 京

内 容 简 介

Visual FoxPro 9.0 是 Microsoft 公司 Visual FoxPro 系列软件的最新版本, 是一种优秀的数据库管理系统和开发工具。本书从易到难, 从简单到复杂, 系统地介绍了 Visual FoxPro 9.0 的使用和数据库应用程序的开发方法。

本书基础知识部分包含第 1 章、第 2 章和第 10 章, 介绍了 Visual FoxPro 开发环境、数据库和表的设计、表单及报表设计、程序调试等基本内容。系统设计部分介绍了 7 个实用而又有代表性的实例开发过程, 这 7 个实例是: 考勤管理系统、客户管理系统、综合人事管理系统、订单管理系统、学位论文管理系统、超市进销存管理系统以及学生成绩管理系统。

本书的特点是: 强调应用性和实战性, 利用实例对 Visual FoxPro 的知识点进行剖析; 注重前后知识关联性, 先独立讲解 Visual FoxPro 中的重要知识点, 然后再将其应用到数据库系统中。

本书适合作为高等院校计算机、自动化、机械、电子等相关专业学生的课程设计指导书, 也非常适合作为开发人员的参考用书。

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 课程设计案例精编/罗颂, 詹斌, 余健编著. —北京: 清华大学出版社, 2008.7

(高等院校课程设计案例精编)

ISBN 978-7-302-17982-5

I. V… II. ①罗… ②詹… ③余… III. 关系数据库—数据库管理系统, Visual FoxPro—程序设计—高等学校—教学参考资料 IV. TP311.138

中国版本图书馆 CIP 数据核字(2008)第 090669 号

责任编辑: 李春明 张丽娜

封面设计: 山鹰工作室

版式设计: 杨玉兰

责任校对: 李玉萍

责任印制: 杨 艳

出版发行: 清华大学出版社

地 址: 北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

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

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者: 北京国马印刷厂

装 订 者: 三河市新茂装订有限公司

经 销: 全国新华书店

开 本: 185×260 印 张: 28.5 字 数: 685 千字

版 次: 2008 年 7 月第 1 版 印 次: 2008 年 7 月第 1 次印刷

附光盘 1 张

印 数: 1~4000

定 价: 46.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题, 请与清华大学出版社出版部联系调换。联系电话: (010)62770177 转 3103 产品编号: 021538-01

前 言

Visual FoxPro 9.0 是 Microsoft 公司 Visual FoxPro 系列软件的最新版本，是一种优秀的数据库管理系统和开发工具。在 Visual FoxPro 9.0 发布以后，SP1、SP2 版本在一年多的时间内相继发布，使 Visual FoxPro 9.0 的功能更加完善，性能更加稳定。Visual FoxPro 功能强大、使用方便，为用户提供了完善的开发环境，是初级软件编程人员理想的开发工具。利用它，用户可以轻松地开发出各种中小型数据库应用系统。本书从易到难，从简单到复杂，系统地介绍了 Visual FoxPro 9.0 的使用和开发方法。

本书的特点是：强调应用性和实战性，利用实例对 Visual FoxPro 的知识点进行剖析；注重前后知识的关联性，先独立讲解 Visual FoxPro 中的重要知识点，然后再将其应用到数据库系统中。

本书分为两部分：基础知识部分和系统设计部分。

基础知识部分包含第 1 章、第 2 章和第 10 章，介绍了 Visual FoxPro 开发环境、数据库和表的设计、表单及报表设计、程序调试等基本内容，还介绍了通过 ActiveX 控件、Windows API 扩展 Visual FoxPro 的知识，以及开发客户/服务器程序等高级应用的知识，最后介绍了如何利用 InstallShield Express 来创建安装程序。所介绍的内容深入浅出、图文并茂，使用了大量的代码进行示例演示，有助于读者掌握 Visual FoxPro 的关键技术。

系统设计部分介绍了 7 个实用而又有代表性的实例开发过程，这 7 个实例是：考勤管理系统、客户管理系统、综合人事管理系统、订单管理系统、学位论文管理系统、超市进销存管理系统以及学生成绩管理系统。

对于以上 7 个实例的介绍方式为：从系统数据库的构建，到设计数据库和数据表、建立各种各样的查询、创建灵活多样的表单、设计形式丰富的报表，进而构造出完整的应用系统。每一个实例都是从头到尾地进行解析。相信读者通过阅读本书不仅可了解如何构建管理信息系统，而且对 Visual FoxPro 9.0 的强大功能会有一个全新的认识。

本书内容深入浅出，循序渐进，为广大读者使用 Visual FoxPro 9.0 进行系统开发设计提供了有力的支持，有助于读者尽快掌握 Visual FoxPro，从而顺利地设计和制作出高质量的应用系统。

本书主要由罗颂、詹斌、余健编写，参与本书编写和程序开发的还有王翠翠、仇亚飞、刘广兴、孙干、侯焕磊、王莹莹、张辰威、柳军旺、孙永全、张伟、寇囡囡、张孝林、陈运来、代小华等。

由于作者水平有限，书中难免有不足和疏忽之处，恳请读者批评指正。

编 者

目 录

第 1 章 Visual FoxPro 的基础知识..... 1	
1.1 Visual FoxPro 简介 1	
1.1.1 向导 1	
1.1.2 设计器 1	
1.1.3 面向对象编程..... 2	
1.1.4 数据库 Rushmore 技术 2	
1.1.5 DBC 事件 2	
1.1.6 SQL 命令的使用 2	
1.1.7 客户/服务器模式 2	
1.1.8 Visual FoxPro 的 Web 功能 3	
1.1.9 智能感知技术..... 3	
1.2 Visual FoxPro 编程基础 3	
1.2.1 基本概念 3	
1.2.2 表达式 10	
1.2.3 函数 13	
1.2.4 结构化程序设计..... 17	
1.3 面向对象程序设计基础..... 23	
1.3.1 对象和类的概念..... 23	
1.3.2 类生成器的基本使用..... 26	
1.3.3 对象事件 29	
1.4 系统开发环境设置..... 30	
1.4.1 显示设置 31	
1.4.2 文件位置设置..... 31	
1.4.3 编辑器设置 32	
1.4.4 表单设置 33	
1.5 数据库设计的一般步骤..... 34	
1.5.1 关系数据库概念..... 34	
1.5.2 建立关系结构模型..... 35	
1.6 小结 36	
第 2 章 Visual FoxPro 的高级应用..... 37	
2.1 数据环境 37	
2.2 表单设计 38	
2.2.1 参数传递与返回值 38	
2.2.2 顶层表单..... 39	
2.3 报表设计..... 40	
2.4 使用 ActiveX 控件 43	
2.4.1 ActiveX 控件概述 43	
2.4.2 ImageList 控件和 TreeView 控件..... 43	
2.4.3 Chart 控件..... 50	
2.5 使用 Windows API..... 53	
2.5.1 如何调用 Windows API 53	
2.5.2 如何防止程序多次开启 54	
2.5.3 如何使窗体透明..... 55	
2.6 开发客户/服务器程序..... 57	
2.6.1 与服务器建立和断开连接 58	
2.6.2 获取或更新数据..... 60	
2.6.3 事务处理..... 62	
2.7 其他开发技巧..... 63	
2.7.1 复制..... 63	
2.7.2 缩进及空行..... 64	
2.7.3 在程序中插入对象名称 65	
2.7.4 精确调整对象位置..... 65	
2.7.5 使用 .prg 文件作为主程序..... 65	
2.8 小结..... 66	
第 3 章 考勤管理系统 67	
3.1 课程设计的目的和意义 67	
3.2 系统设计与分析..... 67	
3.2.1 可行性分析..... 67	
3.2.2 功能描述..... 69	
3.2.3 功能模块设计..... 69	
3.3 数据库设计与实现..... 70	
3.3.1 数据库需求分析..... 70	
3.3.2 数据库表的设计..... 72	

3.3.3 数据库表的实现.....	76	5.2 系统设计与分析.....	159
3.4 系统的实现.....	82	5.2.1 系统功能描述.....	159
3.4.1 系统启动模块.....	82	5.2.2 功能模块划分.....	160
3.4.2 用户登录模块.....	85	5.2.3 系统流程分析.....	161
3.4.3 系统主控模块.....	87	5.3 数据库设计与实现.....	162
3.4.4 基本考勤管理模块.....	89	5.3.1 创建项目和数据库.....	163
3.4.5 加班管理模块.....	100	5.3.2 设计和创建表.....	163
3.4.6 出差管理模块.....	106	5.4 系统的实现.....	168
3.4.7 系统管理模块.....	107	5.4.1 设计项目框架.....	168
3.5 系统运行结果.....	107	5.4.2 部门管理模块.....	172
3.6 小结.....	113	5.4.3 员工信息管理模块.....	179
第4章 客户管理系统.....	114	5.4.4 家庭成员及工作经历 管理模块.....	190
4.1 课程设计的目的和意义.....	114	5.4.5 员工考勤管理模块.....	196
4.2 系统设计与分析.....	114	5.4.6 员工考评管理模块.....	206
4.2.1 功能描述.....	114	5.4.7 用户管理模块.....	211
4.2.2 功能模块设计.....	115	5.4.8 帮助信息.....	218
4.3 数据库设计与实现.....	115	5.5 系统运行结果.....	218
4.3.1 数据库需求分析.....	115	5.6 小结.....	223
4.3.2 数据库表的设计.....	116	第6章 订单管理系统.....	224
4.3.3 数据库表的实现.....	117	6.1 课程设计的目的和意义.....	224
4.4 系统的实现.....	118	6.2 系统设计与分析.....	224
4.4.1 创建项目与数据库.....	118	6.2.1 功能描述.....	224
4.4.2 创建主程序.....	119	6.2.2 功能模块设计.....	225
4.4.3 安全登录模块.....	121	6.3 数据库设计与实现.....	225
4.4.4 主窗口模块.....	125	6.3.1 数据库需求分析.....	225
4.4.5 主菜单设计.....	127	6.3.2 数据库表的设计.....	226
4.4.6 资料管理模块.....	131	6.3.3 数据库表的实现.....	228
4.4.7 投诉管理模块.....	138	6.4 系统的实现.....	230
4.4.8 日程管理模块.....	146	6.4.1 创建可视类.....	230
4.4.9 报表管理模块.....	148	6.4.2 创建系统主表单.....	234
4.4.10 系统维护模块.....	152	6.4.3 用户管理模块.....	239
4.5 系统运行结果.....	155	6.4.4 信息浏览模块.....	255
4.6 小结.....	158	6.4.5 信息管理模块.....	263
第5章 综合人事管理系统.....	159	6.4.6 创建打印输出功能.....	285
5.1 课程设计的目的和意义.....	159	6.5 系统运行结果.....	286

6.6 小结	290	8.4.3 主界面表单模块	351
第 7 章 学位论文管理系统	291	8.4.4 货物管理模块	354
7.1 课程设计的目的和意义	291	8.4.5 进货管理模块	358
7.2 系统设计与分析	291	8.4.6 出货管理模块	362
7.2.1 功能描述	291	8.4.7 报损管理模块	365
7.2.2 功能模块设计	291	8.4.8 库存管理模块	369
7.3 数据库设计与实现	292	8.4.9 用户管理模块	372
7.3.1 数据库需求分析	292	8.5 系统运行结果	372
7.3.2 数据库表的设计	293	8.6 小结	375
7.3.3 数据库表的实现	295	第 9 章 学生成绩管理系统	376
7.4 系统的实现	296	9.1 课程设计的目的和意义	376
7.4.1 创建主文件	296	9.2 系统设计与分析	376
7.4.2 启动表单设计	298	9.2.1 功能描述	376
7.4.3 登录表单设计	299	9.2.2 功能模块设计	377
7.4.4 主表单设计	303	9.3 数据库设计与实现	377
7.4.5 主菜单设计	304	9.3.1 数据库需求分析	377
7.4.6 学生信息管理模块	309	9.3.2 数据库表的设计	378
7.4.7 导师信息管理模块	311	9.3.3 数据库表的实现	381
7.4.8 论文成绩管理模块	314	9.4 系统的实现	383
7.4.9 查询模块	314	9.4.1 公用模块	383
7.4.10 报表模块	319	9.4.2 创建主文件	385
7.5 系统运行结果	328	9.4.3 安全登录模块	386
7.6 小结	333	9.4.4 主表单设计	388
第 8 章 超市进销存管理系统	334	9.4.5 主菜单设计	390
8.1 课程设计的目的和意义	334	9.4.6 学生信息管理模块	395
8.2 系统设计与分析	334	9.4.7 课程信息管理模块	403
8.2.1 功能描述	334	9.4.8 成绩信息管理模块	407
8.2.2 功能模块设计	335	9.4.9 报表管理模块	414
8.3 数据库设计与实现	335	9.4.10 系统管理模块	418
8.3.1 数据库需求分析	335	9.5 系统运行结果	419
8.3.2 数据库表的设计	338	9.6 小结	423
8.3.3 数据库表的实现	342	第 10 章 程序的完善与发布	424
8.4 系统的实现	348	10.1 调试应用程序	424
8.4.1 欢迎界面模块	348	10.1.1 调试器概述	424
8.4.2 用户登录模块	349	10.1.2 查看存储值	426

10.1.3 设置断点	428	10.4 应用程序发布	438
10.2 应用程序的日志管理	429	10.4.1 建立安装项目	438
10.2.1 日志管理实现方法	429	10.4.2 添加文件和文件夹	439
10.2.2 综合人事管理系统 日志管理	429	10.4.3 添加运行时刻组件	440
10.3 创建帮助文件	432	10.4.4 建立快捷方式	441
10.3.1 制作 CHM 帮助文件	432	10.4.5 建立、测试安装程序 和分发应用程序	442
10.3.2 在 Visual FoxPro 中访问 CHM 帮助文件	438	10.5 小结	444

第 1 章 Visual FoxPro 的基础知识

1.1 Visual FoxPro 简介

Visual FoxPro 是一个功能非常强大的数据库应用系统开发工具，本节主要从以下几个方面进行介绍：

- 功能丰富的向导。
- 可视化设计器。
- 面向对象编程。
- 数据库 Rushmore 技术。
- DBC 事件。
- SQL 命令的使用。
- 客户/服务器模式。
- Visual FoxPro 的 Web 功能。
- 智能感知功能。

1.1.1 向导

Visual FoxPro 是一个快速应用系统开发工具(RAD)，它提供了非常丰富的交互式向导，可以帮助用户快速制作各种应用程序对象，包括数据表格、表单、报表、选项卡、查询等。启动向导之后，只需按要求回答一连串的问题，向导就会按照指示建立应用程序对象。如果向导所制作的对象不能完全符合需求，可以随时修改。在使用向导的过程中，可随时按 F1 键，以便获得该向导的联机帮助信息。

1.1.2 设计器

在 Visual FoxPro 中，系统提供了各种各样的可视化设计器(Designer)，这些设计器可以通过交互操作的方式来进行应用程序界面的设计，从而可避免编写大量程序来完成该项功能。所有的设计器都支持快捷菜单，只要在设计器或是内含的控件上右击，便会弹出包含针对该对象各种处理操作的快捷菜单。

系统中常见的设计器及其功能如表 1-1 所示。

表 1-1 Visual FoxPro 中的设计器

设计工具	功 能 用 途
数据库设计器	建立或修改一个数据库以及定义数据表间的关联性连接
数据表设计器	建立或修改数据表以及替数据表编制索引
表单设计器	建立一个用来显示与编辑数据表中数据的表单

设计工具	功 能 用 途
报表设计器	定义报表的格式,以便将数据表中的数据按所需的格式打印出来
选项卡设计器	定义选项卡的格式,以便将数据表中的数据按所需的格式打印出来
查询设计器	设定查询数据需要满足的各项条件,并快速将数据资料查询出来
窗体设计器	用来设置窗体对象以及其中所包含的控件对象

通过使用设计器,建立表单、数据库、数据表格、表单、报表、选项卡或查询的操作过程,都将不再是件痛苦而复杂的工作了。

1.1.3 面向对象编程

Visual FoxPro 不但支持标准的过程化程序设计,而且还在语言上进行了扩展,提供了面向对象编程的强大功能,能够在面向对象程序设计的结构下,重复使用已经规划好的类(Class),快速构建应用程序,提高应用程序的开发效率。

1.1.4 数据库 Rushmore 技术

Visual FoxPro 数据库配合专有的索引技术 CDX(复合索引),只需非常少的时间就可以完成大量的数据存取与查询的操作,其数据索引技术采用 B-Tree 索引技术,这样的存取技术在 Visual FoxPro 中被称为 Rushmore 技术,这项技术对数据库的存取效率的提高有很大的帮助。

1.1.5 DBC 事件

Visual FoxPro 提供了很多 DBC(数据库容器)事件,用户可以编写合适的方法、函数或者过程代码来响应这些事件,以进一步管理数据库。例如控制数据库的打开权限,在数据库的某些操作后附加操作等。

1.1.6 SQL 命令的使用

SQL(Structure Query Language, 结构化查询语言)是关系型数据库通用的结构化查询语言,几乎所有的关系型数据库系统都支持它。SQL 是非过程化的语言,对数据库提供自动导航。它允许用户在更高层次的数据结构下进行工作,可以操作记录集。

Visual FoxPro 的 SQL 命令实际上是一种集成的数据库语言,它能够利用 Rushmore 技术优化性能,并且一个 SQL 命令可以用来代替多个 Visual FoxPro 命令,从而能够方便快速地操作 Visual FoxPro 的数据库。

1.1.7 客户/服务器模式

网络数据库应用系统的一种模式是客户/服务器模式,它符合计算机的开放系统策略和小型化潮流。Visual FoxPro 9.0 为创建功能强大的客户/服务器应用程序提供了一些专用

工具。

Visual FoxPro 客户/服务器模式将 Visual FoxPro 中的功能强、速度快、图形化的用户界面以及高级的查询、报表和处理等优点与严密的多用户访问、海量数据存储、内置安全性、可靠的事务处理和日志以及 ODBC 数据源或服务器的本地语法等功能紧密地结合在一起, 为用户提供功能强大的客户/服务器解决方案。

1.1.8 Visual FoxPro 的 Web 功能

随着网络技术及应用的不断发展, 数据库对网络环境的支持也不断地增强。在 Visual FoxPro 中, 不仅支持原来的 Web 发布功能, 而且增加了 Web 服务功能和对 XML 的支持。

1.1.9 智能感知技术

Visual FoxPro 系统提供了新的智能感知(IntelliSense)功能, 这些功能能够自动地填写声明、属性、参数, 可减少程序员必须输入的代码数量, 并且程序员无须再去参考文档来查找所需的参数, 从而使代码的编写更加容易、方便, 如图 1-1 所示。

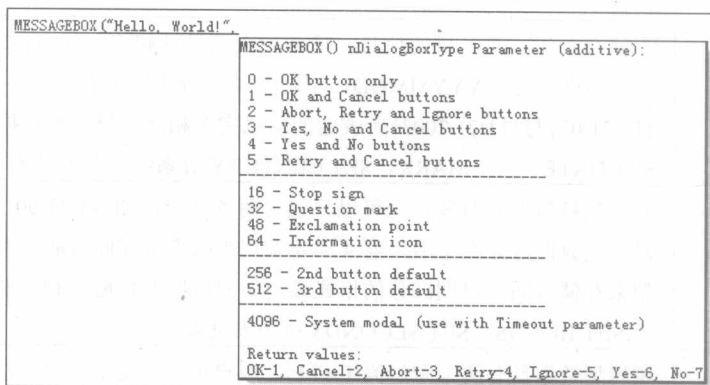


图 1-1 IntelliSense 技术

通过以上的介绍, 可以看出 Visual FoxPro 是一个功能非常强大的数据库应用系统开发工具。

1.2 Visual FoxPro 编程基础

Visual FoxPro 提供了一套方便实用的数据库管理开发语言, 由命令和函数两部分组成。本节重点讨论变量、数据类型、表达式和系统常用函数等系统开发的语言基础以及结构化程序设计的相关知识。

1.2.1 基本概念

在学习 Visual FoxPro 的编程知识时, 首先需要了解 Visual FoxPro 的一些基本概念,

比如数据类型、常量、变量和数组等。这些概念在各种程序语言中都会遇到，是掌握程序语言的基础。

1. 数据类型

数据类型是数据的基本属性，它描述了系统可接收数据的类型和范围，也确定了对数据进行的操作和运算。一般只有相同类型的数据之间才能进行运算，在操作中必须随时注意操作对象的数据类型。

Visual FoxPro 程序设计中的数据类型如表 1-2 所示。

表 1-2 Visual FoxPro 的数据类型

数据类型	说 明
Character(字符型)	由字母、数字、空格、符号、标点和汉字等组成，在定义时要用引号(在 Visual FoxPro 中使用单引号和双引号均可)
Currency(货币型)	当需要保存有关货币计量的数据时，使用货币型数据。货币型数据可以自动控制小数点后的位数，当所赋予数值的小数点后位数超过 4 时，VFP 将在运算之前将其自动作舍入处理。货币型数据的格式可以使用 SET CURRENCY 命令来设置
Date(日期型)	存储有关年、月、日的信息，通常用大括号括起来，如{11/21/03}。在内存中的存储格式为 YYYYMMDD，年份占 4 个字节，月份和日各占 2 个字节，即{11/21/03}与{11/21/2003}所表达的信息完全相同。日期型数据的格式可以使用 SET DATE、SET MARK、SET CENTURY 等系统命令来设置
DateTime(日期时间型)	同时包括日期和时间，也可以只包含两者之一，如{11/21/00 12:00AM}。日期时间型数据占 8 个字节，前 4 个字节用整数型来存储日期，后 4 个字节用整数型来存储时间。日期部分的格式设定如同日期型数据，时间部分的格式可以使用 SET HOURS、SET SECONDS 命令来设置
Logical(逻辑型)	逻辑型数据具有两个布尔值.T和.F.。用户定义的变量、数组在默认的情况下都是逻辑型，值为.F.
Numeric(数值型)	表示一个整数或者实数，在内存中占 8 个字节，取值范围为 -0.9999999999E+19~0.9999999999E+20
Object(对象型)	作为一个对象的名称，唯一地表示该对象。可以使用一些特殊的命令来创建，如：Myform1=CREATEOBJECT("FORM")
Varient(不定型)	能存储任意类型的数据。一旦把数据赋值给一个不定型的变量，该变量的类型即变为该数据的类型

可以使用 Type()函数返回指定表达式的数据类型，如：

Type(字符表达式)

其中，“字符表达式”可以是变量、常量、字段等其他表达式，表达式必须作为字符串传递，因此常量、变量和字符表达式一般要用双引号引起来。

测试表达式的数据类型的命令语句为：

```
?TYPE("34.7")
```

```
N
```

表示返回值 34.7 的数据类型为 N(数值型)。

2. 常量

常量(Constant)是一个固定的数据项,或者是在操作过程中不能改变的数据。常量可分为字符型常量、数值型常量、货币型常量、日期型常量、日期时间型常量及逻辑型常量。所有的字符型常量都必须用一对引号括住;所有的货币型常量必须在前面加上一个货币符号(\$);所有的日期型常量与日期时间型常量都必须用大括号括住;所有逻辑型常量都必须用圆点(.)括住。

3. 变量

变量是进行一种算法时可以用来存储值的容器,是允许在操作过程中取不同值的数据元素。Visual FoxPro 的变量名最长可由 254 个字符组成,变量名可以是字母、数字、符号或汉字,但是绝对不能与命令名、函数名等系统保留字相同。另外特别要注意,变量名最好不要与使用中的数据表名、字段名相同。

变量是程序的基本单元,在 Visual FoxPro 中,变量分为字段变量、内存变量、系统变量和对象变量 4 种。

(1) 字段变量

字段变量是数据库中定义的变量。字段变量属于永久变量,其数据类型包括字符型、数值型、日期型、逻辑型、备注型和通用型等。数据表的每一个字段都是一个字段变量。字段变量的值只能使用 REPLACE、UPDATE 命令来更改。

 **提示:** 如果字段变量与内存变量同名,而字段变量又没有采用表别名作为前缀时,系统将先检验当前工作区中的字段变量,若没有这样的字段再检查内存变量。如果希望 Visual FoxPro 优先考虑内存变量名,应当在此变量前加上引导字符 M.或 M->。例如: M.NAME 或 M->NAME。

(2) 内存变量

内存变量是一种临时变量,是在程序或者命令状态时用于存放临时数据的内存工作单元,它在每次清除内存或退出 Visual FoxPro 时自动释放。内存变量的数据类型包括字符型、数值型、日期型和逻辑型等。

Visual FoxPro 设置的内存变量根据作用域一般分为局部变量(Local)、私有变量(Private)和全局变量(Public)。

- **局部变量(Local 关键字):** 所声明的变量或数组只能在创建它们的程序中使用,不能被更高层或者低层的程序使用。当程序执行完毕后,该变量会直接从内存删除。
- **私有变量(Private 关键字):** 所声明的变量或数组可以在下一个程序中继续使用,尤其适用于程序文件彼此之间的调用。在上层程序定义私有变量后,可在下层程序中直接引用;但是下一层程序所定义的私有变量无法被上一层程序引用。

- 全局变量(Public 关键字): 所声明的变量或数组对于整个应用程序都是可用的。如果在应用系统的任何位置设置了全局(公有)变量, 可以被应用系统中任何程序、对象事件引用。它会一直存在于内存中, 直到应用系统结束, 或者用户通过特别指令加以释放。

 **提示:** 如果变量没有使用作用域关键字声明, 而直接定义并赋值, 系统默认为局部变量。

内存变量的命名约定规则为: 一般以字母(汉字也可)或者下划线开头, 由字母、数字和下划线组成, 至多 128 个字符, 不可与系统保留字同名。

为了增加程序的可读性, 便于多人阅读和开发, Visual FoxPro 对内存变量的命名规定如下: 变量由两个小写字母开头, 第一个表示作用域, 第二个表示数据类型; 后面为变量名称。表 1-3 和表 1-4 分别列出了变量作用域和数据类型的前缀字符的用法。

表 1-3 变量作用域的前缀字符

作用域	前缀	示例
局部	l	lcName
私有	p	pnTel
全局	g	gnNumber

表 1-4 变量数据类型的前缀字符

数据类型	前缀	示例
数组	a	paArray(8)
双精度型	b	lbGrade
字符型	c	lcName
日期型	d	pdDate
浮点型	f	lfTotal
通用型	g	lgPic
整数型	n	gnNumber
对象型	o	poMyApp
日期时间型	t	ptDateTime
货币型	y	pyCurrency

(3) 系统变量

系统变量是系统内部定义的一些变量。这些变量的名称是系统已经定义好的, 都以字符“_”(下划线)开头。所以在定义内存变量名称时, 最好不要以字符“_”开头, 以避免重名。

 **提示:** 系统变量的操作可遵循内存变量的规则, 但是不能使用内存清除命令如 CLEAR MEMO 或 CLEAR ALL 清除这些变量。

(4) 对象变量

Visual FoxPro 是面向对象的语言，所以，系统提供一种称为对象的变量。对象变量是一种新型变量。

下面介绍有关内存变量的操作。对于内存变量，系统提供了各种操作命令。

(1) 内存变量赋值

可利用 STORE 命令或运算符“=”给变量赋值。语法为：

STORE <表达式> TO <内存变量表> | <数组表>

或：

<内存变量> | <数组> = <表达式>

表达式可以表示任何数据类型。

例如将字符串“Visual FoxPro”赋值给变量 mystring 的命令语句为：

STORE "Visual FoxPro" TO mystring

或：

mystring = "Visual FoxPro"

 **提示：** 如果字符串本身就包含单引号 (') 或双引号 (")，请以下列三种方式之一来引用此字符串。

- 将字符串包含在一对中括弧中，如：

STORE [I'm a student.] TO mystring

- 如果字符串本身包含单引号，可将字符串包含在一对双引号中；如果字符串本身包含双引号，可将字符串包含在一对单引号中。如：

STORE " I'm a student." TO mystring

- 如果字符串本身既包含单引号又包含双引号，则必须将字符串包含在一对大括号中。

(2) 内存变量输出

内存变量输出的语法为：

? | ?? <表达式 1> [AT<显示列的位置>]

?：表示打印输出，最后回车换行。

??：表示不回车换行。

AT<>：表示列位置坐标。

该命令首先计算变量或表达式的值，然后按照指定的格式将结果显示在屏幕上。

(3) 内存变量内部结构显示

即分屏显示当前所有内存变量内部结构的相关信息。语法为：

DISPLAY MEMORY LIKE 框架

或：

LIST MEMORY LIKE 框架

框架指匹配格式，可以包括通配符“*”（与任意字符匹配）和“？”（与一个字符匹配）。

内存变量内部结构显示命令的命令语句为：

```
A=1
AB="Visual FoxPro"
ABC=.F.
DEF="YOU ARE A STUDENT"
G=678
DISPLAY MEMORY LIKE A?
```

主窗口显示运行结果：

```
A      Pub      N      1      (      1.00000000
AB     Pub      C      "Visual FoxPro"
```

在上述 DISPLAY MEMORY 命令的输出结果中：

第一列为变量名。第二列为变量特性。若为全局变量，则显示 Pub；若为局部变量，则显示 Priv。第三列为变量类型。字符型显示 C；数值型显示 N；日期型显示 D；逻辑型显示 L。第四列为变量的值。对于字符型变量，显示带引号的字符串；对于数值型变量，显示数值，其后括号里是机内表示法；对于日期型变量，显示格式为“月月/日日/年年”（可使用 SET DATE TO 命令改变其显示格式）；对于逻辑型变量，显示.T.或.F.。

(4) 清除内存变量

清除内存变量就是将已定义的某些内存变量或数组从内存中除去，被清除的内存变量将释放占用的内存区，以后不能再使用。语法为：

```
CLEAR ALL      && 此命令会清除内存中所有的变量
RELEASE 变量名表
```

或：

```
RELEASE [ALL LIKE 框架 | ALL EXCEPT 框架]
```

框架指匹配格式，可以包括通配符“*”和“？”。

- ALL LIKE 框架：指定保存所有满足指定框架的内存变量和数组。
- ALL EXCEPT 框架：指定保存所有不满足指定框架的内存变量和数组。

此命令会清除指定的变量，或清除符合框架格式的变量。命令语句为：

```
RELEASE AA, AB      &&清除、释放变量 AA 与 AB
RELEASE ALL         &&清除所有变量
RELEASE ALL LIKE A*  &&清除开头字母为 A 的所有变量
RELEASE ALL EXCEPT A* &&清除开头字母不为 A 的所有变量
```

 **提示：**“&&”是注释符号，符号后面的字符均为注释语句；“*!”也是注释符号，但是只能另起一行注释，不能跟在命令语句的后面。如果命令行中某一行的结尾有“;”，表示下一行的内容为该行内容的后续。

(5) 内存变量存盘

如果需要长期保存内存变量和数组，可以将当前内存中的变量和数组作为一个扩展名为 .MEM 的内存文件或备注字段保存在外存储器中。语法为：

```
SAVE [ALL LIKE 框架 | ALL EXCEPT 框架]
TO 内存变量文件名 | TO 备注字段名
```

内存变量文件名：指定保存内存变量和数组的内存变量文件。内存变量文件的默认扩展名是 .MEM。

备注字段：指定保存内存变量和数组的备注字段。

(6) 恢复内存变量

恢复保存在内存变量文件或备注字段中的内存变量和内存变量数组，并把它们放回到内存中。请注意对象类型变量不能存入内存变量文件或内存变量备注文件。语法为：

```
RESTORE FROM FileName | MEMO MemoFieldName [ADDITIVE]
```

ADDITIVE：防止删除当前内存中已有的内存变量或内存变量数组。

如果使用 ADDITIVE 时，要添加的内存变量或内存变量数组的数目加上已有内存变量的数目超过了系统对内存变量数目的限制，Visual FoxPro 将从内存变量文件或备注字段中恢复尽可能多的内存变量和内存变量数组。恢复内存变量或内存变量数组时，如果内存变量或内存变量数组与已有内存变量或内存变量数组有相同的名称，则用恢复的内存变量或内存变量数组的值改写原有内存变量或内存变量数组中的值。如果命令中没有包括 ADDITIVE 关键字，RESTORE FROM 将清除当前内存中的所有内存变量或数组。RESTORE FROM 命令不影响系统内存变量。

 **提示：** 当在程序中执行 RESTORE FROM 命令时，所有 PUBLIC 和 PRIVATE 内存变量和数组都恢复成 PRIVATE 变量或 PRIVATE 数组，所有 LOCAL 内存变量和数组都恢复成 LOCAL 变量或 LOCAL 数组。如果在命令窗口中执行 RESTORE 命令，则所有 PUBLIC 和 PRIVATE 内存变量和数组都恢复成 PUBLIC 变量或数组，所有 LOCAL 内存变量和数组都恢复成 LOCAL 变量或数组。

在下面的示例中，先创建两个内存变量，然后把它们保存到一个内存变量文件中，随即又恢复到内存中。所有这些操作并没有清除原来已有的内存变量。命令语句为：

```
gnval1 = 1
gcval2 = ' Visual FoxPro '
save to temp
clear memory
gdval3 = date( )
restore from temp additive
clear
display memory like g*
```

4. 数组

数组是按一定顺序排列的一组内存变量，数组中的各个变量称为数组元素，数组必须