

数据库与管理信息系统 实验教程

苏选良 张益星 林荣清 等编著

一本集Visual FoxPro管理信息系统

和计算机综合作业处理于一体的配套实验教材



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

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

数据库与管理信息系统实验教程

苏选良 张益星 林荣清 等编著

電子工業出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

本书为 Visual FoxPro 和管理信息系统实验教程, 是管理信息系统课程的配套用书。该书集 Visual FoxPro、管理信息系统和计算机综合作业处理实验于一体, 通过精选的实验来诠释和演绎面向对象的程序设计, 循序渐进, 能较快而有效地提升程序设计能力。全书分为基本控件和综合实验两部分。书中还提供各类实用的表单模型, 帮助读者理解现实的管理过程, 也提供了最基本的设计文档。最后, 还提供了一些标准的例程, 供读者举一反三, 灵活掌握。

本书适合于本科、高职高专学生以及从事管理信息化的工程技术人员阅读。

未经许可, 不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有, 侵权必究。

图书在版编目 (CIP) 数据

数据库与管理信息系统实验教程 / 苏选良等编著. —北京: 电子工业出版社, 2004.2

ISBN 7-5053-9590-4

I. 数… II. 苏… III. ① 数据库系统—教材 ② 管理信息系统—教材 IV. ① TP311.13 ② C931.6

中国版本图书馆 CIP 数据核字 (2004) 第 001362 号

责任编辑: 孙延真 特约编辑: 李双庆

印 刷: 北京彩艺印刷有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

经 销: 各地新华书店

开 本: 787×1092 1/16 印张: 19 字数: 480 千字

印 次: 2004 年 2 月第 1 次印刷

印 数: 5000 册 定价: 29.00 元

凡购买电子工业出版社的图书, 如有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系。

联系电话: (010) 68279077。质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

序 言

《数据库与管理信息系统实验教程》一书是作者继《管理信息系统》一书之后，为完善数据库与管理信息系统课程的实践性教学环节而编写的一本配套教材，填补了数据库与管理信息系统课程实验教学方面的空白点。该书按应用型本科建设的要求和思路，以由浅入深，循序渐进的方式引导学生进行程序设计，构建了一个从单项练习，综合练习到标准例程的完整实验体系，帮助学生进一步巩固理论教学的效果，更多地了解开发工具，提高学习兴趣，让学生在理论与实践的结合中找到信心与成就感。

该书结构新颖、实例丰富、文字简练、附录实用，紧密结合教学实践，饱含教学经验。相信在学习与实践中，读者会进一步认识本书的特点与价值。

刘迎春
2004 年 1 月
于湖南工程学院

前 言

亲爱的读者，这是一本能有效提升程序设计能力并集 Visual FoxPro、管理信息系统和计算机综合作业处理于一体的实验教材。在这里，我们将为你打开程序设计的魔方；在这里，我们将使你曲径通幽；在这里，你会找到信心、兴趣和价值。帮你找到信心，激发兴趣，让你尽快进入角色，正是本书的目标。在练习前，你不必阅读那些厚厚的 Visual FoxPro 参考书，当你完成这些练习，或许一气呵成，或许经历种种“磨难”时，你已经成为一个具有相当设计经验和能力的程序设计者，我们相信你能达到这样的目标。

许多同学视程序设计为畏途，常常一筹莫展，无从下手，甚至把程序设计看成是高深的东西，是程序设计师们的事情，往往把自己遇到的障碍归结为技术不好。如果你是初学者，也许你会想：难道程序设计真的离我很远吗？我也能成为程序设计师吗？就潜能而言，或许你不比任何人差，只是没有很好开发而已。程序设计确实需要经验和智慧，你可能正缺乏这样的经验，但这并不妨碍你进行程序设计，任何优秀的程序设计员也都有一个积累的过程。其实，程序设计最重要的不在于你了解的命令和函数的多少，而在于你对事物逻辑的把握和表达。在面向对象的程序设计中，可能真实情形同你想像的恰好相反，程序设计其实是一件简单的事情，甚至比 DOS 时代简单得多，所以，你不要先入为主，更不要轻易地怀疑自己的能力。

本书可作为 Visual FoxPro 和管理信息系统实验配套教材，也可作为计算机综合作业处理课程的教材，还可以作为 Visual FoxPro 等级考试机试参考书。但本书不是 Visual FoxPro 教科书，如果你真是一片空白，也大可不必惊慌，本书将帮你速成，当然你比别人要辛苦一点。如果没基础，浏览 Visual FoxPro 教科书也是必要的，但也未必切合你的需要，或许你没有充分的时间把一本厚厚的书系统读完，或许即使读完，也未必马上知道该从哪里入手。本书虽然篇幅不长，但比较实用并物有所值。本书系统地介绍了 Visual FoxPro 设计的各种技巧，并且由浅入深地启发你的兴趣。如果你想较快地熟悉 Visual FoxPro 的程序设计，请阅读有关章节并按本书的指引去练习！

特别感谢全国应用型本科教育协作组副组长单位，湖南工程学院副院长刘迎春教授为本书作序并给予指导。参加本书编写的除了苏选良还有湖南工程学院计算机科学与技术系的张益星、郭芳和福建经济管理干部学院的林荣清。湖南工程学院经济管理系的梅益军等也参与了本书的编写（因署名所限未能列入）。本书是在讲义的基础上，结合教学实践修订而成的。第 1 稿讲义由林延朝整理，图文由袁凡编辑、整理，在此一并致谢！因时间和水平的局限，可能存在一些错误，敬请读者批评指正并在修订时更正、完善。

作 者

于湖南工程学院

2003 年 11 月

目 录

第 1 章 Visual FoxPro 程序设计的基本概念.....	(1)
第 2 章 Visual FoxPro 的主要设计器及技巧.....	(8)
第 3 章 Visual FoxPro 实验中的常见错误及处理.....	(14)
第 4 章 Visual FoxPro 控件和基类对象实验.....	(20)
第 5 章 Visual FoxPro 综合表单实验.....	(58)
第 6 章 面向应用的表单参考模型.....	(176)
第 7 章 Visual FoxPro 学习资源.....	(194)
附录 A Visual FoxPro 的基类.....	(283)
附录 B 核心事件语法与功能.....	(283)
附录 C 常见控件及功能.....	(284)
附录 D 常见对象及功能.....	(285)
附录 E 常见事件及功能.....	(285)
附录 F 常用方法及功能.....	(287)
附录 G 常用函数和命令一览表.....	(288)
附录 H 主要数据表结构.....	(289)
附录 I 系统概要设计说明书编写规范.....	(290)
附录 J 系统详细设计说明书编写规范.....	(294)

第 1 章 Visual FoxPro 程序设计的基本概念

“面向对象”这词听起来抽象，理解起来更抽象，加上派生出来的一堆新名词，更是叫人丈二和尚摸不着头脑，但如果不理解这些概念，在面向对象程序设计(OOP)中，你将寸步难行。怎么办呢？那就只能化抽象为具体了。本书将以通俗的语言诠释这些概念，以感性的方式将你引入 OOP 的世界，相信在感性和实践中去领悟这些概念比费力的解释要好得多。

一、走进 Visual FoxPro

大多数高校都开设 XBase 系列数据库，考虑到知识的继承性和学习基础，我们选择 Visual FoxPro 作为实验和课程设计环境，这能较好地完成实验和设计任务，同时也能与社会需要相吻合。要完成课程设计任务，需要掌握 Visual FoxPro 的设计方法。如果以前没有直接学过 Visual FoxPro，实验中可能有一些困难，但不要放弃。

Visual FoxPro 是一个在 Windows 下使用的面向对象的数据库，也是目前国内流行的数据库(DB)软件之一，它是在 FoxPro 2.x 基础上发展起来的，用 32 位指令系统编写的 PC 数据库软件。目前广泛使用的中文版本是 Visual FoxPro 6.0。Visual FoxPro 是在 XBase 基础上逐步发展起来的，它的发展脉络是：dBase→FoxBase→FoxPro 2.x→Visual FoxPro 3.0→Visual FoxPro 5.0→Visual FoxPro 6.0。Visual FoxPro 与 XBase 的分野是 Visual FoxPro 3.0。从 Visual FoxPro 3.0 开始数据模型从关系模型发展到关系加对象模型，引入了面向对象技术，使用了类和继承、封装、多态特性、方法程序和事件模型，提供了大量可视化编程对象等，完全的事件驱动和基于图形用户界面(GUI)。Visual FoxPro 真正进入客户/服务器体系结构，可以方便地存储、检索和处理服务平台上的关键信息，访问与开放数据库互连(ODBC)兼容的 DB 中的文件，所以 Visual FoxPro 3.0 作为第 4 代语言，对 FoxPro 而言，是一个里程碑式的产品，是真正的面向对象数据库(ODDB)，使软件生产效率提高到易使用和易维护的新水平。

Visual FoxPro 6.0 中文版又有了许多新的增强与改进。新增了组件管理库、新的向导、新的语言元素、项目管理器连接、Access 和 Assign 方法程序等。增强和改进了对 Web 的支持、对象链接与嵌入(OLE)的拖放、Automation 服务程序、Active Document、HTML 帮助创建、对图形交换格式(GIF)和联合图像专家组(JPEG)的支持和代码范围分析器应用程序。

Visual FoxPro 6.0 提供了集成的开发环境来提高应用开发的效率。Visual FoxPro 6.0 提供了表设计器、数据库设计器、表单设计器、报表设计器、菜单设计器、查询设计器和类设计器等来完成项目的某一单元，最后通过项目设计器这一设计框架集成各设计单元的成果，整合成系统。在开发环境中，表单是界面设计的主要领域。通过可视化地定义对象属性比通过程序定义可提高编程效率。

二、表单的世界

回想 FoxPro 2.x 的程序设计，都是通过命令窗口（Modify Command 命令）来编辑.prg 命令文件，但在 Visual FoxPro 中已经有了很大的不同。虽然.prg 命令文件仍可以按同样的方式建立并运行，但已经不再是 OOP 程序设计的主流模式了，只是在项目编译时才用到一个很短的引导程序 main.prg,其他的程序设计几乎都让位给对象的事件代码或自定义的方法程序了。因为表单是可以放置各种对象的容器，表单上的每一个对象又有自己的事件和方法，为驱动对象的事件去实现其预定的功能，需要为对象的事件和自定义方法编程，而包含数据与操作的对象又统一于表单文件中，几乎绝大多数的编程都是在表单里完成的。因此，表单是 OOP 程序设计的主要领域，工具栏和菜单程序终将归结于此，其他设计结果的运行也无不与表单相联系。因此要弄清楚 OOP 设计，重点要理解表单。那么，什么是表单呢？

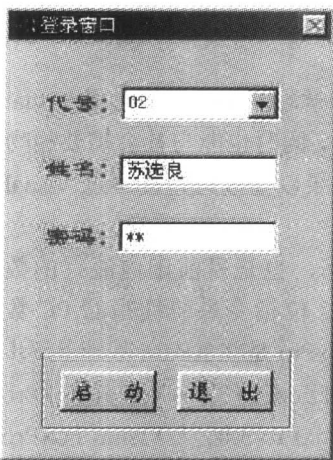


图 1.1 登录表单示例

简而言之，表单就是窗口(window)，即视窗，所以它是新概念，但不是新东西。因为它与 Windows 其他窗口没啥不同，是一种统一的用户界面。Windows 下的各种窗口在 Visual FoxPro 中就可称之为表单。表单也是一种能包容其他对象的容器，就好比是菜篮子，里面可以盛各种蔬菜。在 Visual FoxPro 中，放在表单这个菜篮子里的对象就是各种控件，关于控件的概念将在后面介绍。下面我们就以图 1.1 为例（一张登录表单）来认识一下具体的表单吧。

图 1.1 是一个登录窗口，该表单包含了 9 个基本对象：即 3 个标签对象，1 个组合框对象，两个文本框对象，两个命令对象和一个形状对象。3 个标签对象分别为：代号、姓名、密码，用来提示信息；组合框用来选择操作代号；两个文本框用来输入相应的姓名和密码。由此不难看出表单不仅是窗口，也是能容

纳别的对象的容器类对象。在表单对象上可以放置控制对象，而无法在控制对象上放置表单对象，如只能将鸡蛋放入菜篮子，而不能将菜篮子放在鸡蛋上一样。请记住表单文件扩展名是.scx，还有一个伴侣——即存放表单数据的备注文件.sct，这两个文件可是成双成对，形影不离的，如果“分居”，这个表单就执行不了了。如果主文件名不一致，表单也不执行。这样的情形还有不少，将在后面予以介绍。关于表单的更多应用与技巧，我们将在设计器部分进行较详细的说明。

三、基本概念

我们将结合 Visual FoxPro 设计环境来介绍几个基本概念，如控件、属性、事件、方法和类等。

1. 控件

表单实际上就是一个用户界面，用户通过这个界面来完成数据录入的工作，而控件就是这个界面的各个组成部分，即对象，实际上是表单上的对象实例，是放在表单上用以显示数据，

进行数据操作或使表单更易于理解和操作的一种图形对象。如图 1.1 表单用来输入用户姓名的文本框 text1 即是控件。控件是从哪里来的呢？控件的发源地是表单控件工具栏，如图 1.2 所示。Visual FoxPro 如同其他开发环境一样，不仅提供了 19 个控件、13 个内部对象以及 4 个对象参考，也提供了第三方控件的使用途径（OLE 控件），正如使用基类控件一样，将大大拓展你的设计能力。第三方控件都是现成的，拿来使用，既省力又省时。如果把控件比做是鸡蛋的话，那么表单控件工具栏就是会生蛋的母鸡。这个“生蛋”的过程其实非常简单，只要鼠标轻轻一点就可完成。用鼠标选中你所要的“母鸡”——基类控件（以形象的图标表示，只要你将鼠标停在某图标下，就会显示该控件类的名称），然后释放在表单的适当位置。即使最一般的智力，学会“生蛋”也不要 5 分钟时间。“生蛋”过程好比搬运，运输工具就是鼠标，货源地就在表单控件工具栏，目的地在表单，因此“生蛋”的过程（初步设计过程），也就是将控件从表单控件工具栏运到表单的过程。下面我们看图 1.2，来认识一下这个“生蛋”的“母鸡”吧！

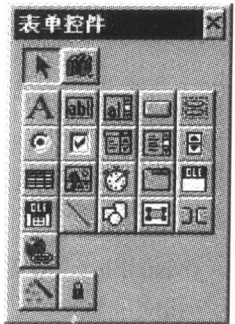


图 1.2 表单控件
工具栏

图 1.2 中包括了标签、文本框、编辑框、命令按钮及按钮组、选项按钮组、复选框、组合框、列表框、微调按钮、表格、页框等，你需要花 10 分钟熟悉这些基本控件，它们都有许多自己的属性、事件和方法。在面向对象的应用程序设计中，用户通过对象的属性、事件和方法程序来处理对象。关于这些，将在后面的控件练习中介绍。不可能也没必要了解全部的属性、事件和方法，只要把握重点就行。在“属性窗口”移动鼠标时，相应的中文说明就出现在该窗口的底部，没必要去死记，因此不必为记这些东西而犯愁。

2. 属性



图 1.3 属性窗口

所谓对象的属性，指的是对象表现出来的外部特征。一个属性就是对对象的一项描述内容，用以描述该对象的特质，如文本框 text1 的 visible 就是用来设置该文本框是否可视。属性设置的工具是属性窗口，就好像一部电话有一定的外观颜色和大小，而它的听筒也有拿起和放下两种状态。我们可以把电话的外观颜色和大小，听筒的两种状态称之为电话的属性，而在 Visual FoxPro 中每个对象也都有自己的属性，例如控件在表单中的定位位置和尺寸大小(Left、Top、Width、Height)，提示文字 (Caption 属性)，名字 (Name 属性)。不要混淆 Caption 属性和 Name 属性。除了通过属性窗口定义属性以外，还可以在程序中设置属性，但一定要注意数据的类型，对字符型数据要加引号，逻辑型数据更要注意表达方式。用程序方式定义时，所有属性定义均要用等号，属性值也可用表达式。下面是对象属性定义窗口的式样，如图 1.3 所示。

记住可以为表单定义新的属性，方法是通过菜单“表单”→“新建属性”添加属性就可以了，要注意添加完毕后要关闭、退出。添加完后就可以在表单中看到新添加的属性。如果因重复添加而出错，就要删除已定义的属性再重新添加。同样可以在程序中完成这一操作。当在实验中出错，找不到属性时，就要检查一下表单的属性，看是否有此属性，如没有就要添加，其界面如图 1.4 所示，添加一个新的 oktoleave 属性。

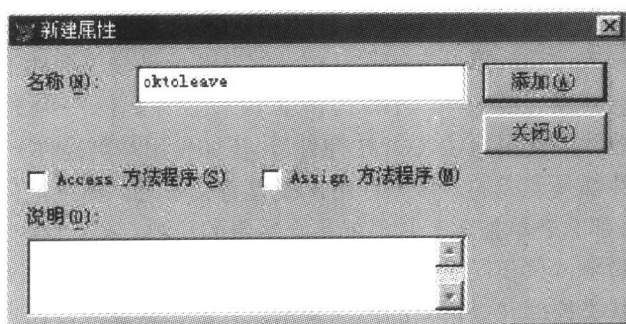


图 1.4 添加新属性窗口

3. 事件与方法

Visual FoxPro 是一种完全的事件驱动，所谓的事件驱动就是当触发对象的某一事件时，就执行相应的指令，即程序代码。所谓对象的事件是一种预先定义好的特定动作，能够被对象本身识别，如某对象鼠标的单击事件（即 Click 事件），就好像电话的拨号接通操作。而对于在表单中的按钮，用户可以单击，该单击动作相对于按钮这一对象而言，就是一个可以识别的事件，如同家用电器上的按钮一样。显然事件应该具有与之相关联的方法程序。再以电话为例，设计电话的工程师们必须为电话设计一套内部电路，当用户拿起电话机拨打了一个电话号码后（事件发生后），这套电路开始工作——处理用户拨号，并完成与所拨打电话的连接。这套内部电路相当于一系列的动作指令，在这里就是为事件编写的代码序列，也就是事件触发时所要做的处理，即产生的动作结果。对于事件驱动程序而言，要得知用户的动向，对于必须响应的事件需编写该事件的响应程序。通过表单设计器编写事件代码很简单，只需双击某对象就可以进入代码编辑窗口。图 1.5 就是为“取消”按钮编写单击事件的代码窗口，一定要注意选择对象与过程，这两个组合框很重要，不可随便取默认值。

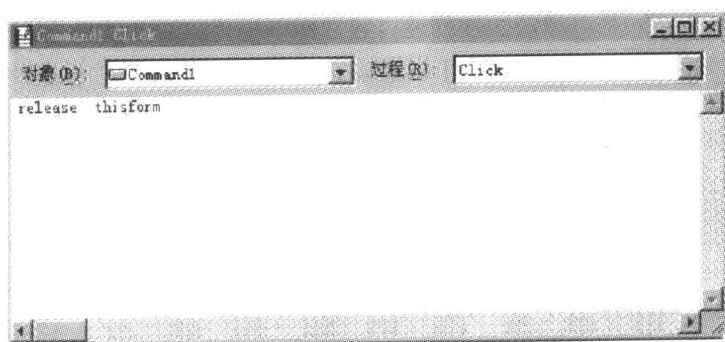


图 1.5 代码编辑窗口

当然，也可以通过命令窗口以.prg 文件的形式来编写事件代码或方法程序，但一定要注意加过程结构 `procedure~endproc`。一定要注意与代码编辑窗口的区别。仍以上面的“取消”按钮为例进行比较说明，如图 1.6 所示。

图 1.6 只表达了事件结构 `command1.click` 这一部分，一个完整的 OOP 风格的.prg 文件还需要创建对象实例、定义类和添加对象。创建对象实例可以用表单的 `CreateObject()` 方法或 `NewObject()` 方法。定义类可以用 `Define Class...As...With~Enddefine` 语句。向容器添加对象并定义属性可用 `Add Object...As...With;` 语句或 `AddObject()` 方法。这是在.prg 文件中必须使用的

OOP 程序的基本结构。看懂这些结构，就能构造出基本的 OOP 命令文件。可以看出事件代码与方法程序就是以结构化程序设计为基础，再加上面向对象的表示，这就是 OOP 与 SP 在事件代码或方法程序上的统一。下面参考一个 OOP 命令文件实例，注意加粗部分。

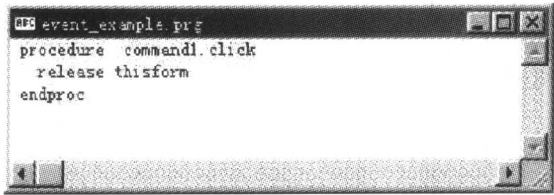


图 1.6 .prg 文件编辑窗口

```

** 代码和程序结构如下:
PUBLIC oform1
oform1=NEWOBJECT("form1")
oform1.Show
RETURN
*****
** Form:          form1
** 父类:  form
** 基类:  form
DEFINE CLASS form1 AS form
    Top = 0
    Left = 0
    Height = 156
    Width = 283
    DoCreate = .T.
    Caption = "实例"
    Name = "Form1"
    ADD OBJECT check1 AS checkbox WITH ;
        Top = 96, ;
        Left = 96, ;
        Height = 25, ;
        Width = 73, ;
        Caption = "背景风格", ;
        Name = "Check1"
    ADD OBJECT label1 AS label WITH ;
        AutoSize = .T., ;
        FontName = "隶书", ;
        FontSize = 20, ;
        Caption = "欢迎使用本表单", ;
        Height = 33, ;
        Left = 46, ;
        Top = 36, ;
        Width = 198, ;
        BackColor = RGB(255,0,0), ;
        Name = "Label1"
```

```

PROCEDURE Init
    thisform.label1.BackStyle=0
ENDPROC

PROCEDURE check1.InteractiveChange
    if this.value=1
        thisform.label1.BackStyle=1
    else
        thisform.label1.BackStyle=0
    endif
ENDPROC
ENDDEFINE
*****

```

只要把握了上面的程序结构，在.prg 文件中编写 OOP 程序就不再是一件难事了。弄清这个模板，就能举一反三。特别注意在类结构中，过程结构之外不容许插入任何 ForPro 程序命令，它会当成类成员来处理。也要特别提醒在设计前要想一下是为哪个对象的哪个事件编写代码，这是我们一直在强调的问题。否则，写了很多代码，可能完全写错地方，而没有任何正确结果，所以要擦亮双眼，不要乱点鸳鸯谱，记住选择对象和过程。

在面向对象的程序设计中，可以为对象的每一可识别的事件设计好相应的程序，当针对该对象的某一事件发生后可执行对应的事件程序，完成事先设计好的任务。所谓的事件程序，亦即对象的方法程序，设置对象的各个属性并为对象的每一事件编程，这就是面向对象的程序设计的基本思想。而在 Visual FoxPro 中，表单是面向对象程序设计的主要领域，对菜单项的编程和对工具栏的编程，也最终体现为表单程序设计。因此表单设计过程，实际上就是面向对象的程序设计过程。但不同于过去接触的.prg 文件，而现在的程序设计完全是事件驱动，并且对象及其属性，事件与方法程序是封装在一起的，没有离开对象的属性，也没有离开对象的事件或方法，这是一个最基本的规则，也是与.prg 程序的主要区别。

事件与方法虽然过程结构相同、程序结构相同，但两者是有区别的。方法是一个为处理某特定事件而编写的一段程序，它是为某一事件而存在的，它限于被封装的对象调用，从这些角度观察方法就像是一个过程或自定义函数，方法可以被调用，但本身属于特定的对象。事件是给定的，既不能多，也不能少，没有自定义的空间，表单可以自定义方法。新方法的定义基本类似于新属性的定义，即新建方法，过程如图 1.7 所示，即在菜单中选择“表单”→“新建方法程序”→“添加”→“关闭”。

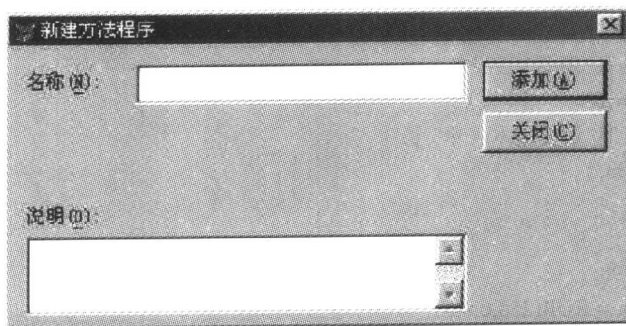


图 1.7 新建方法窗口

这里需要强调哪些是系统方法，而哪些是自定义方法。如果是系统给定方法就不需劳神费力了，只要懂得调用就行，如 `thisform.show()`。对于特定的需要而自定义的方法，就需要按上面的方式新建方法并在代码编辑窗口自定义方法程序，这同编辑事件代码完全一样。方法同事件的另一个不同点就是方法可以带参调用（参数写在括号里），如上面的 `thisform.show()` 方法。

4. 类

类是生活中一个很普通的概念，我们常说“物以类聚，人以群分”，也把一些很特殊的群体称为“另类”。我们常把一些具有相同特征的对象归结在一起而成为一类，称为分类或合并同类。从这些常识不难看出，类是具有相同属性的对象的集合，如文本框类产生的所有对象都具有文本框的一切特征，只是取值可能不同而已。类好比是模具，前面提到的“表单控件工具栏”就是这种产生对象的模具，工具栏里的每一类就是其中的一副模具。Visual FoxPro 提供了 20 余个内部类别和基类（包括容器类和控件类）。

类和继承性是现代软件工程的两个重要概念，也是两个相互关联的概念。类的好处就在于它的继承性，也就是当遇到类似的情形时，可以继承父类的属性和方法或少许修改或添加新的数据或程序便可产生继承父类一切特性的新的子类或衍生类别，而无需从头开始，可以充分利用现有资源而形成新的资源，因此类和继承的意义在于知识保留和资源积累，这种积累是微软成功的基础，你可以看到类的巨大作用了吧。类可以积累资源，提高设计效率，这是软件开发不能忽视的资源。

类的文件扩展名是 `.vcx`，类的产生可以通过类设计器来实现。在介绍设计器时再进行详细说明。类也可以通过程序方式来实现，途径是 `Newobject()` 方法和 `Define Class...As...~Enddefine` 结构，注意加粗部分，举例如下：

```
myform=Newobject("frmclass")    &&产生新的对象类别
myform.show()

Define   Class frmclass As form    &&定义该类的属性
    Caption="我的标题"
    Left=100
    Top=30
Enddefine    &&结束类定义
```

除了上面的方法以外，可直接用命令来建立类，即：

Create Class <类名>或

Create Class <类名> of <类库名>

可直接用命令在程序中打开类库：

Set CLASSLIB To <类库名> Additive

理解这 5 个概念是非常重要的，花很多篇幅来阐述也是基于其重要性。上面提到的结构与模板也是不可忽视的。在此基础上了解 Visual FoxPro 程序设计的两个特质：对象参考和面向对象的表示法（圆点号：表示对象与子对象、与属性、与事件、与方法的关系），就了解了 OOP 程序设计的核心，再加上对结构化程序设计的充分了解，就可以畅游于 OOP 程序设计之中了。在下面一章，将介绍主要的设计器。

第 2 章 Visual FoxPro 的主要设计器及技巧

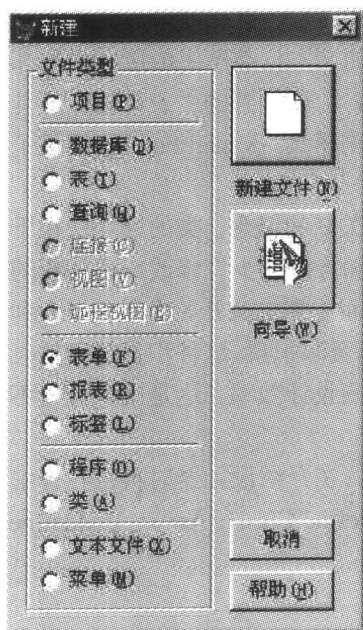


图 2.1 各设计器入口表单

前面已对表单进行了初步讨论，那么表单从哪里产生呢？很显然，表单设计器就是产生表单的工具。那么如何进入表单设计器呢？在启动 Visual FoxPro 之后，单击 Visual FoxPro 主窗口中工具栏上的第一个按钮——“新建”按钮，进入了汇集各种设计器的表单，即各设计器入口，如图 2.1 所示。

一、表单设计器

请记住：这就是设计的大门，通过这里可以进入各设计器。先以表单为例，在选项组中选择“表单”，即可通过“新建文件”或“向导”进行表单设计了。如图 2.2 就是通过向导来做的。

向导就像傻瓜相机，通过向导来完成表单设计快速而简洁，但仍然不鼓励一开始就使用向导，虽然很容易，但学不到真东西，也不一定符合个性化的需要。在这里，我们不讨论表单向导，而通过“新建文件”进入空白表单模板。下面我们将通过图形方式进入表单设计器模板，见图 2.3。

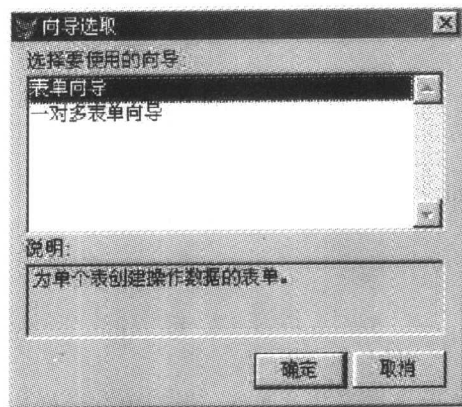


图 2.2 表单向导选择窗口

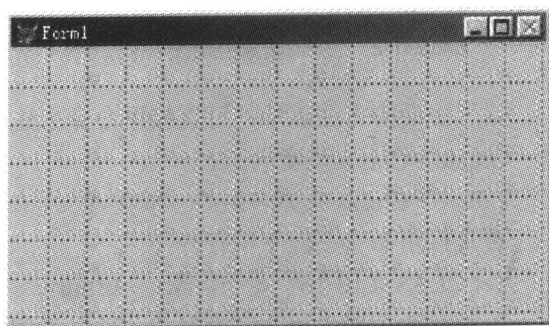


图 2.3 表单设计器模版

这个时候，我们已经进入了表单模板，即表单设计器了，窗口上同时出现了表单工具栏和属性窗口，如果在操作时两者不见了，不必着急，在菜单“显示”里，单击“属性”或“表单控件工具栏”，两者又会出现眼前了，这在前面已讲过，不重复。万事俱备，现在可以进行

表单设计了。表单设计其实可以分为以下 4 个方面。

1. 根据应用的需求确定表单功能、对象及事件

根据应用的需求确定表单功能、对象及事件是最重要的也是最容易被忽略的。回到前面的图 1.1 登录表单，它的功能就是用来判定用户的身份，分别需要对操作代号，用户名，口令进行检验并根据检验结果做出相应的选择。所谓检验就是将这些用户输入值同预设值进行比较，只有这 3 个值同时满足条件，才判定身份合法。在此过程中，要考虑到用户的一些需求，如口令修改。同时，要给出必要的提示信息，甚至要对破译口令的企图予以必要的限制，这与银行取钱时的情形很类似。这些预设值究竟应放在哪里呢？从安全和多用户的角度看，这些值应放在表中，并对口令表中的数据做加密处理。概括地讲，登录表单的功能就是对用户身份的合法性进行判断，判断的方式就是输入值与预设值的比较。只要确定表单的功能也就确定了表单对象属性及事件，所以表单设计中最重要设计，就是功能设计，这只能根据业务的需要来确定表单的数据及处理方式，不可能有统一的模式。你可能要去采集原始信息单据并加以格式化处理，以适合计算机处理。从这个角度看，应用表单的设计关键在于对业务或商业逻辑的了解，这不是一个技术问题，而是对管理的理解问题。

2. 初步设计

初步设计很简单，就是将第一步确定的对象放入表单中，方法在前面已介绍，即从表单控件工具栏→表单，在表单控件工具栏中单击需要的对象基类图标至凹下后，将鼠标移到表单中，光标呈“十”字形，这时松开鼠标左键，对象就产生了，并修改相应属性，如标签，命令按钮的“Caption”属性等，直到完成。当所有对象定义好后，就进行一些布局处理，尽可能使表单美观、方便，这个过程虽然不需要太花精力，但确实要有替用户着想之心，还要有点美术细胞，除此之外，就没有什么难度了。

3. 程序控制逻辑设计

程序控制逻辑设计令许多人望而却步。当需要独立完成一项设计时，你可能茫然一片，不知如何下手，甚至不知道做什么，怎样做，即不知道要为哪个对象的哪个事件编码。这需要强化第一个步骤，明晰功能，即有哪些功能需求，需要为哪个对象的哪个事件编码，最怕乱点鸳鸯谱或囫圇吞枣，知其然而不知其所以然。OOP 中容易犯的最大错误就在于搞错对象，搞错事件或两者同时弄错，张冠李戴。语法错误属基础问题，而对象、事件错误属概念性问题。当然进行 OOP 设计需要一定的基础，如了解 FoxPro 2.x 中的主要命令和函数，程序设计的 3 种基本结构。如果思路不是很清晰，最好画一张程序流程图，对减少逻辑性错误很有帮助，这都是传统的 FoxPro 2.x 的设计模式，在 OOP 中仍有生命力，但在 OOP 中要注意 OOP 的特质，及 OOP 与 SP（结构化程序设计）的统一。在 Visual FoxPro 中，过去的许多命令如今都已转化成对象，我们不在此举例，但在控件实验中将说明这种联系。

你也许看不懂 `thisform.text2.value`，想不到这在图 1.1 中就是用户输入的口令值。如果去掉左右空格，就是输入的纯口令值，即 `allt(thisform.text2.value)`。其实 Visual FoxPro 的精华就几乎表现在这句话里，也可以说这就是 Visual FoxPro 的特质。这句话表达的含义就是当前表单下的文本框 2 的输入值，即用户输入值。特质之一就是对象参考，犹如英语中的代词，指代当前某对象。这样的对象参考其实不多，就 4 个，它们是：

Thisformset	当前表单集
Thisform	当前表单
This	当前对象（不是指表单，而是表单中包含的对象）
Parent	当前对象的父对象

用得较多的是后面 3 个，在实验中经常遇到，尤其要注意 This 和 Parent 的用法，Thisform 用法简单，，一般不会出错。特质之二就是一个不经意的圆点号(.)，几乎浓缩了 Visual FoxPro 的精华，反映了对象与子对象、属性、事件、方法的隶属关系。表示如下：

O.OI→表示对象与子对象的关系

O.P→表示对象与属性的关系

O.E→表示对象与事件的关系

O.M→表示对象与方法的关系

具体表示如下表：

关 系	示 例	含 义
O.OI	Thisform.text2	表示当前表单下的文本框 text2
O.P	Text2.value	文本框 text2 的值
O.E	Command1.click	命令按钮 command1 的单击事件
O.M	Form1.show()	表单 form1 的显示方法,运行当前表单

对象事件与方法的共同之处：结构都为：procedure～endproc，都需要编程，都是属于某一特定对象。不同之处：事件不能增，而方法可以自定义，可增，方法与过程、函数、子程序相当。如何自定义方法在前面已介绍。结构化设计(SP)加面向对象的表示方法就是 SP 与 OOP 的统一，所以如果熟悉 SP 的基本结构，只需熟悉 OOP 表示法就可以进行 OOP 设计了。

4. 驱动和连接

请注意以下驱动方式：

序 号	驱 动 方 式	驱 动 对 象	驱 动 位 置
1	Do form <表单名>	表单	对象事件
2	Do <命令文件名>	程序	命令窗口
3	Do <菜单文件名>.mpr	菜单	表单 Load/Init 事件
4	Report form <报表文件名> Preview/Printer	报表	对象事件或菜单项
5	Command.click	工具栏	对象事件

在对象事件中驱动表单（包括在菜单和工具栏中）均采用第 1 种方式。而在表单中驱动菜单，则可以在表单的 Load 事件或 Init 事件中启用第 3 种方式。

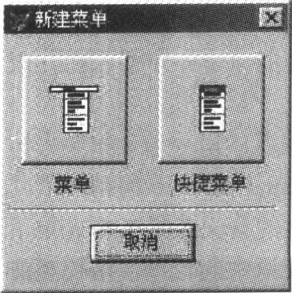


图 2.4 菜单选择窗口

二、菜单设计器

菜单设计器的进入方式如同表单，同样可以通过工具栏上“新建”图标选择菜单进入。菜单设计有两种情形，一种是通用横条菜单，而另一种是快捷菜单，菜单选择窗口，如图 2.4 所示。

快捷菜单是通过对象的右键单击触发，不触发的时候，看不到

菜单，而横条菜单不管是否触发总出现在表单上部。横条菜单在表单中驱动，而快捷菜单在表单包含的对象的右击事件中驱动，这是两者的差别，但都是同样的文件格式.mnx，备注文件为.mnt。两种菜单生成同样的菜单程序文件.mpr，生成方式是菜单编完后在菜单设计器的“菜单”选项中单击“生成”便生成了程序文件。要查看和修改菜单程序文件，可以在命令窗口通过 modify command <菜单文件名>.mpr 就可完成。在菜单中驱动表单如同执行命令，在表单中驱动菜单，需要在表单的 Load 事件或 Init 事件中运行菜单程序文件，记住一定要生成程序文件.mpr，同时注意不漏掉文件的扩展名.mpr，否则会视同为命令文件。

三、报表设计器

如同其他设计器一样，报表设计的工具是报表设计器。报表设计器的使用非常简单，但要弄清报表对象并弄清报表各带区的细节，就要多花工夫，尤其在报表数据的处理和报表格式定义方面要用心去思考。报表的输出是通过调用 Report form <报表文件名> printer/preview，其中 printer 子句是将输出结果送往打印机，而 preview 子句是将输出结果送到屏幕，即预览，只能二选一。当然报表输出前要判定文件是否存在，要选择输出方式，如果是打印，还要选择所支持的打印机及判定设备是否连接好，并给出提示信息，这牵涉到设备驱动、状态检测，当然这已经超出了本书的范围，需要在实践中进一步摸索，并参考有关的书籍。要记住报表的文件名是.frx，报表备注文件名是.frt，也是成双成对的。

进入报表设计如同表单，也有两种方式，即“新建文件”和“向导”，如果选择“新建文件”则直接进入了报表设计器，如果选择“向导”，它会带你一气呵成。选择“向导”又分为“报表向导”和“一对多报表向导”，如图 2.5 所示。

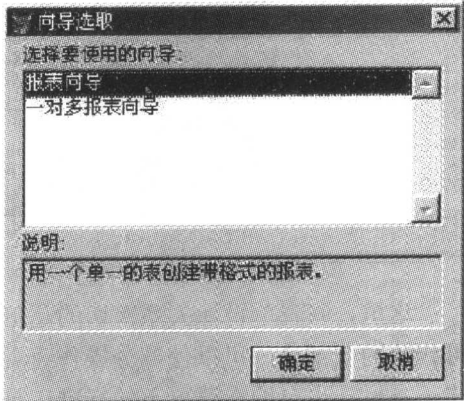


图 2.5 报表向导选择

四、数据库设计器和表设计器

Visual FoxPro 中的数据库概念与 FoxPro 2.x 中的数据库概念完全不同，弄清这种区别很重要。在 FoxPro 2.x 中数据库指的是自由表.dbf 文件，而 Visual FoxPro 中的数据库是表上面的一个层次，它是数据库表、视图及关联的集合，Visual FoxPro 中表分为数据库表和自由表，数据库表不同于自由表，要打开数据库表必须先打开数据库，然后才能打开表。我们可以将自由表加入数据库而成为数据库表，也可以从数据库中取出数据库表而成为自由表。这些可通过设计器也可通过命令来实现，可以通过前面的方式进入数据库设计器，也可以通过命令方式(Create Database)进入数据库设计器。要注意视图是一种虚表，即逻辑表，它可以像表一样操作，本身并不独立存在，必须依附于表，如同索引。但视图为数据检索提供了极大方便。视图分为本地视图和远程视图，视图可以通过视图设计器完成，视图的本质是 SQL 查询。关联就是过去的主子表之间的关系，过去通过 set relation to 语句来完成，现在可通过连接设计器或 relation 对象来完成，不在此单独介绍了。要记住数据库文件扩展名为.dbc，备注文件扩展名为.dbt。