

Visual Basic 6 24学时学习教程

(美) Greg Perry 著
Sanjaya Hettihewa

刘双才 曾庆敏 张宗涛 等译

康博创作室 审校

机 械 工 业 出 版 社
西蒙与舒斯特国际出版公司

本书作为Visual Basic 6 自学教程,向读者介绍了Visual Basic初级或中级程序员需要掌握的全部知识。内容包括集成开发环境介绍、基本控件使用、语法规则、文件操作、调试技术、ActiveX控件、系统对象、应用程序发布以及网络知识等。在内容安排上,由简单到复杂,读者无需懂得太多的背景知识。通过每学时的实例一步步引导,读者能很快掌握Visual Basic 6 的编程方法,并可积累一定的编程技巧。全书分为24学时,每学时可以用一小时或更少的时间学完,通过24学时的学习,读者将能够学会编写一般的Windows应用程序。

本书适合Visual Basic初学者和具有一定编程基础的程序员。

Greg Perry & Sanjaya Hettihewa:Sams Teach Yourself Visual Basic 6 in 24 hours.
Authorized translation from the English language edition published by Sams Publishing.
Copyright1998 by Sams Publishing.
All rights reserved.For sale in Mainland China only.

本书中文简体字版由机械工业出版社和美国西蒙与舒斯特国际出版公司合作出版,未经出版者书面许可,本书的任何部分不得以任何方式复制或抄袭。

本书封底贴有Prentice Hall防伪标签,无标签者不得销售。

版权所有,翻印必究。

本书版权登记号:图字:01-98-1915

图书在版编目(CIP)数据

Visual Basic 6 24学时学习教程/(美)佩里 (Perry,G.), (美)海蒂希娃(Hettihewa,S.)
著;刘双才等译. —北京:机械工业出版社,1999
书名原文:Sams Teach Yourself Visual Basic 6 in 24 Hours
ISBN 7-111-06969-2

I.V… II.①佩… ②海… ③刘… III.BASIC语言—教材 IV.TP312

中国版本图书馆CIP数据核字(98)第36007号

出版人:马九荣(北京市百万庄大街22号 邮政编码100037)

责任编辑:廉一兵

北京昌平第二印刷厂印刷·新华书店北京发行所发行

1999年1月第1版第1次印刷

787mm×1092mm1/16·17印张

印数:0001-6000册

定价:45.00元(附光盘)

凡购本书,如有倒页、脱页、缺页,由本社发行部调换

前 言

您大概急于开始这本24学时的Visual Basic学习教程。但在开始的时候应花上些时间熟悉一下本书的内容结构，以下几部分将对此做介绍。

谁应该看本书

本书是为那些希望尽快掌握Visual Basic的程序员和将要成为程序员的人士编写的，读者可以不必具有该语言的基础知识。Visual Basic是一种适于很多水平人士使用的编程语言，从未编过程序的初学者学完两个学时后将能编写出完整的Windows程序，就像本书所演示的那样。那些使用过其他编程语言的人也将会喜欢用Visual Basic来进行程序设计，使得创建一个Windows程序更像是设计一个具有鼠标驱动的艺术程序的屏幕。

本书在几个层次上讲解Visual Basic。读者可迅速地按照里面的简单例子开始设计应用程序。这些应用程序将完全是Windows的应用程序，它们具有所有的常用用户界面控件，例如命令按钮、标签和文本框。

只要读者很熟悉Windows，那么就可以创建Visual Basic应用程序。您不必是一位Windows专家，但应用熟练地使用菜单、鼠标以及Windows界面。如果读者已经可以打开、关闭和重新设置窗体的大小，那么您肯定具有创建Visual Basic应用程序的必要能力。

这本24学时教程讲解的是Visual Basic 6，这是最新、最好的Visual Basic版本。Visual Basic 6要求Windows 95/98或是Windows NT 4.0。Windows 95的用户界面(现在在Windows NT中也出现)使用户可以在一个舒适的窗口化环境进行工作。

本书将为您做些什么

虽然这不是一本参考书，但读者将可以确切地学到一个初级或中级程序员用来创建有用的、强大的、有趣的应用程序所需要知道的东西。高级技术细节是大多数程序员用不上的，本书没有收录它们来浪费读者的时间。读者可能很想在24学时内尽快开始编写Visual Basic程序，本书可以实现您的目标，并可使您积累一些编程技巧。

本书介绍了一个新的Visual Basic程序员所需要的背景知识和理论。除了讨论背景知识之外，本书还很实用，它提供了很多任务使得读者可以一步步创建Visual Basic应用程序。随着读者学习教程的进展，这些任务就从简到繁，逐步增加。

本书真的能在24学时之内教会读者使用Visual Basic 6吗？

是的。读者可以在一小时或更少的时间内，掌握一学时的内容(这样，本书的每一学时可以看作是“课时”)。虽然有些学时的内容较长，但是对内容材料做了均衡。较长的课程中包含几个任务，而较短的课程中包含有背景材料。内容材料是由任务、背景知识和深层的解释以及帮助学习的提示来进行均衡组织的。这些使得在学习本书的每一页上都有新鲜感。

您需要些什么？

本书需要读者拥有一台Windows95兼容机并安装有Windows95/98或NT。另外，还需要安装Visual Basic 6.0。如果已拥有安装Windows95/98(或Windows NT)以及Visual Basic 6的硬件的话，那么您需要做的是使用本书并学习Visual Basic编程。

Visual Basic光盘上的文件

有些学时涉及到图形文件，这些文件是可以作为Visual Basic 安装的一部分被安装的。为了在硬盘上找到这些文件，必须在安装Visual Basic时选择安装图形文件。读者或许可以在D:\Program Files\Microsoft Visual Studio\Common\Graphics\文件夹中找到这些图形文件。如果不能在该文件夹中找到图形文件，那么可以使用Windows来寻找该文件夹。选择Start\Find\Files or Folders。搜索本地硬盘以及Visual Basic光盘来查找VB图形目录中的Graphics。

本书约定

每一学时中出现新术语时随后都会作出解释，而问答部分放在每一学时的最后以加深巩固所学的内容。另外，各学时都采用测验和练习的形式来进一步巩固所学的内容。

本书采用了几个共同的约定来帮助讲解编程内容。下面给出本书编排上的一些约定：

- 如果一个任务要求从一个菜单中进行选择，本书将用一个垂直条把菜单项分开。所以，本书使用File\Save As来表示读者应该打开File菜单再选择Save As命令。
- 当学习一种编程语言时，读者经常需要学习一个命令的语法或格式。与下面几行类似的语句将帮助读者学习一个新的Visual Basic语言命令：

For CounterVar=StartVal To EndVal [Step IncrementVal]

Visual Basic语句块。

Next CounterVar

正体字符文本表示读者将要输入到程序中的代码(编程语言信息)，如For和Next，表示读者必须准确输入的关键字。中括号内的信息表示可选代码，如果需要，则可以敲入，比如[Step IncrementVal]。

除了编排上的约定外，以下的特殊标志用来表明不同的信息，以便于识别：

新术语 当一个新的术语第一次出现时，读者会发现一个黑体的“新术语”标志和定义，这样可强调该术语。

注意 “注意”特别解释每一学时所阅读的内容。它们对概念和程序进行细致的解释。

技巧 读者将发现许多技巧。这些技巧提供了一些捷径和解决普遍问题的方法。

警告 “警告”部分提醒读者易出现的错误。阅读它们将节省时间并避免麻烦。

如何与作者联系？

关于作者Sanjaya,读者可查找以下的Web地址：<http://www.NetInnovation.com/Sanjaya/>。读者也可以通过e-mail与他联系，他的e-mail地址如下：Sanjaya@NetInnovation.com。

目 录

前言

第一部分 Visual Basic简介

第1学时 开始学习 Visual Basic	1
1.1 Visual Basic内容	1
1.2 语言	2
1.3 Visual Basic的各种版本	3
1.4 VB编程过程	3
1.5 启动Visual Basic	3
1.6 退出Visual Basic	5
1.7 掌握开发环境	5
1.8 帮助系统	9
1.9 小结	9
1.10 问答	9
1.11 复习	10
第2学时 分析Visual Basic程序	11
2.1 事件驱动程序	11
2.2 控件事件	12
2.3 分析应用程序示例	12
2.4 命名对象	14
2.5 运行应用程序	14
2.6 程序代码	15
2.7 事件过程	17
2.8 属性和事件过程	17
2.9 从头生成应用程序	18
2.10 小结	19
2.11 问答	19
2.12 复习	20
第3学时 控件和属性	21
3.1 创建新的应用程序	21
3.2 控件提供界面	22
3.3 给用户以帮助	25
3.4 命名文本型数据	27
3.5 休息一下	28
3.6 小结	30
3.7 问答	30

3.8 复习	31
第4学时 查看标签、按钮和文本框	32
4.1 理解运行时控件焦点	32
4.2 鼠标和热键不需要焦点	33
4.3 相关的属性	34
4.4 Tab顺序	34
4.5 命令按钮	34
4.6 标签	36
4.7 文本框	38
4.8 窗体属性	40
4.9 小结	40
4.10 问答	41
4.11 复习	41

第二部分 编写代码

第5学时 把代码放入Visual Basic中	43
5.1 编写代码的基本概念	43
5.2 数据基本概念	45
5.3 表达式和数学运算符	51
5.4 小结	53
5.5 问答	53
5.6 复习	54
第6学时 消息框和输入框	55
6.1 函数概述	55
6.2 MsgBox()和InputBox()	56
6.3 Visual Basic的Code窗口帮助	59
6.4 一个捷径:添加注释	60
6.5 InputBox()详解	62
6.6 小结	63
6.7 问答	63
6.8 复习	63
第7学时 判断	65
7.1 比较运算符	65
7.2 If语句	66
7.3 If语句的Else分句	68
7.4 用逻辑运算符进行复合比较	69

7.5 用Select Case语句进行多重选择	70	11.5 VB的时钟:Timer控件	112
7.6 另外两个Select Case格式	72	11.6 小结	113
7.7 小结	74	11.7 问答	113
7.8 问答	74	11.8 复习	113
7.9 复习	74	第12学时 对话框基础知识	115
第8学时 Visual Basic的循环	76	12.1 公共对话框的作用	115
8.1 Do While循环语句	76	12.2 添加公共对话框控件	118
8.2 Do Until循环	78	12.3 生成公共对话框	120
8.3 其他Do循环	79	12.4 公共对话框的方法	120
8.4 For循环	80	12.5 添加File(文件)对话框	121
8.5 小结	83	12.6 Color对话框	121
8.6 问答	84	12.7 Font对话框	121
8.7 复习	84	12.8 Printer对话框	122
		12.9 小结	122
		12.10 问答	122
		12.11 复习	122
第三部分 如何使用代码		第四部分 用数据编程	
第9学时 将代码和控件组合起来	85	第13学时 模块化编程	125
9.1 利息计算应用程序	85	13.1 结构化编程	125
9.2 使用控件数组	87	13.2 调用过程并返回	126
9.3 完成窗体	89	13.3 编写子例程代码	128
9.4 添加代码	89	13.4 编写函数代码	129
9.5 Unload语句	91	13.5 编写参数代码	131
9.6 完善显示形式	91	13.6 通过引用和值接收参数	132
9.7 错误检查	92	13.7 小结	133
9.8 小结	95	13.8 问答	134
9.9 问答	95	13.9 复习	134
9.10 复习	96	第14学时 节省时间的内置函数	135
第10学时 列表框和数据列表	97	14.1 许多函数	135
10.1 列表框控件	97	14.2 数值函数	135
10.2 组合框	101	14.3 字符串函数	136
10.3 数据数组	102	14.4 日期和时间函数	137
10.4 控件数组	103	14.5 数据测试函数	140
10.5 小结	104	14.6 数据转换函数	140
10.6 问答	104	14.7 格式函数	141
10.7 复习	105	14.8 小结	143
第11学时 其他控件	106	14.9 问答	143
11.1 选项按钮	106	14.10 复习	143
11.2 框架和选项按钮	107		
11.3 复选框	110		
11.4 滚动条	111		

第15学时 Visual Basic数据库基础	145	19.3 完成工具栏	193
15.1 文件概念	145	19.4 Line和Shape控件	195
15.2 数据库简介	151	19.5 小结	199
15.3 Data Form Wizard	155	19.6 问答	199
15.4 小结	156	19.7 复习	200
15.5 问答	157	第20学时 编写正确的应用程序	201
15.6 复习	157	20.1 错误种类	201
第16学时 用Visual Basic打印	159	20.2 调试器	204
16.1 打印简介	159	20.3 设置断点	205
16.2 为用户准备打印	160	20.4 单步执行代码	207
16.3 介绍Printer对象	161	20.5 Call Stack显示运行所处位置	208
16.4 Print方法	163	20.6 Immediate窗口	209
16.5 开始打印	166	20.7 小结	210
16.6 小结	168	20.8 问答	210
16.7 问答	168	20.9 复习	211
16.8 复习	168	第六部分 高级Visual Basic应用程序	
第五部分 修饰程序		第21学时 Visual Basic和ActiveX	213
第17学时 菜单和Visual Basic	171	21.1 ActiveX:用户使用的工具	213
17.1 Menu Editor	171	21.2 用VB创建ActiveX控件	215
17.2 给应用程序增加菜单栏	173	21.3 OLE 过程	217
17.3 为菜单选项命名	174	21.4 ActiveX文档	218
17.4 增加下拉菜单选项	175	21.5 小结	220
17.5 菜单的其他元素	177	21.6 问答	221
17.6 将菜单与事件过程连接起来	178	21.7 复习	221
17.7 在工程间拷贝菜单	179	第22学时 对象基础	222
17.8 小结	180	22.1 系统对象	222
17.9 问答	180	22.2 程序对象	223
17.10 复习	180	22.3 使用集合和对象数组	224
第18学时 图形图像控件	182	22.4 介绍OLE自动化	227
18.1 Image控件	182	22.5 小结	230
18.2 Picture Box控件	185	22.6 问答	230
18.3 动画图片	186	22.7 复习	230
18.4 小结	189	第23学时 发布应用程序	232
18.5 问答	189	23.1 编译应用程序	232
18.6 复习	190	23.2 设置项目属性	232
第19学时 工具栏及其他图形	191	23.3 建立应用程序	234
19.1 准备工具栏	191	23.4 启动Package And Deployment Wizard	235
19.2 Image List控件	192		

VIII

23.5	运行安装	237
23.6	卸载应用程序	237
23.7	小结	237
23.8	问答	238
23.9	复习	238
第24学时 联机 Visual Basic		239
24.1	跟随向导上网	239
24.2	用户需要一个ISP	241
24.3	使用浏览器	241

24.4	查看其他工具	244
24.5	小结	244
24.6	问答	244
24.6	复习	245

第七部分 附 录

A	运算符优先顺序	247
B	答案	248
C	使用本书配套光盘	262

第一部分 Visual Basic简介

第1学时 开始学习Visual Basic

欢迎学习Visual Basic!你拥有了目前最强大的和轻松的Windows开发工具。通过24学时的学习,读者将会觉得Visual Basic确实很有用。即使读者几乎没有或从未开发过Windows应用程序,只要你熟悉Windows环境,在使用Visual Basic时也会感到随心所欲。通过本学时的学习,读者将会变成对Visual Basic 6.0的大部分内容很熟悉。

本学时的主要内容包括:

- Visual Basic能做什么
- 如何启动Visual Basic
- 如何停止Visual Basic
- 什么时候使用不同的Visual Basic窗体
- 窗体和系统如何一起工作

1.1 Visual Basic内容

Microsoft Visual Basic 6.0是旧BASIC语言的最新的和最好的版本,它在一个软件包内为用户提供完全的Windows应用程序开发系统。Visual Basic(或VB,通常这样称它)可使用户编写、编辑并测试Windows应用程序。此外,VB包含了可以编写和编辑帮助文件、ActiveX控件(在第21学时中有其内容),甚至Internet应用程序(在第24学时中有其内容)的工具!

新术语 控件是工具箱窗口中的工具,可将其放到窗体中与用户交互以及控制程序流程。

新术语 程序就是一个指令集,指令可使计算机做例如计算等事情(程序通常是应用程序的替代词)。

Visual Basic本身就是一个Windows应用程序。加载或执行VB系统与处理其他Windows程序是一样的。可以用运行的VB程序来创建其他程序。虽然VB是一个极为强大的工具,但是它仍然是一个程序员用来编写、测试以及运行Windows应用程序的工具。

新术语 项目是用户创建的组成Windows应用程序的文件集合。

尽管程序员通常将项目与应用程序交替使用(就像本书所用的那样),但应用程序这种说法最适合描述一个Windows程序,因为Windows程序一般是由几个文件组成的。这些文件组成在一起形成一个项目。项目产生最终的程序,用户可以通过双击图标或选择Windows开始菜单中的应用程序来加载并运行Windows下的这个最终程序。

新术语 应用程序是可编译为可执行文件的一个或多个文件的集合。

随着计算机硬件的发展,编程工具的作用已经发展演化了45年时间。今天的一个编程工具比如Visual Basic,与几年以前的编程语言有了很大的不同。在Windows环境出现之前,编程语言是一个简单的基于文本的被用来编写程序的工具。今天需要比只是一种语言更多更多的东西;需要一个图形开发工具,它可以在Windows系统中运行并利用Windows所提供的全部的图形、多媒体、在线和多任务功能来创建应用程序。Visual Basic就是这样一个工具。不仅是一种语言,Visual Basic可以创建出对当今Windows运行系统每一个方面进行交互的应用程序。

注意 虽然Visual Basic是一个综合性的编程工具,但VB仍保留其BASIC语言的内容。设计者在50年代后期开发出的BASIC编程语言是为初级编程人员使用的。BASIC曾是那时比其他语言(比如COBOL和FORTRAN)更为容易使用的编程语言。Microsoft在发展Visual Basic时,从未忘记VB的根。编程的初学者可以在很短的时间里编写出一个简单可行的Windows程序。在下一学时结束以前,读者将会使用Visual Basic来编写Windows程序。

新术语 向导(Wizard)是问答对话框,它可以自动执行任务。在本书的各处,将会使用向导来协助编写Visual Basic应用程序。

新术语 编译器是一个系统,它将用户所编写的程序转换为一个计算机可执行的应用程序。

如果读者在以前已经看过Visual Basic,那么将会对如今的Visual Basic系统感到吃惊。VB现在运行的是一个真正可以创建独立运行时的.exe文件的编译器,它比以前快得多。VB还包括几个向导,能提供逐步渐进的对话框问题来指导用户创建应用程序。VB的开发平台是一个被称为Developer Studio的开发环境,现在支持像Visual C++和Visual J++那样高级编译器的特性功能。当学会一种Microsoft的可视编程工具之后,那么将具有无需走太多弯路就能使用另一种语言的能力。

新术语 Developer Studio是Visual Basic的开发环境。

1.2 语言

今天的编程语言与以前不一样了。语言本身并非减少了重要性;而图形界面对应用程序来说已经成为更加重要的内容。

计算机不能理解任何一个人所说的语言。像意大利或英语这样的口语对于计算机来说由于过于普通和模糊而不能使其理解。因此,我们必须适应机器并学会一种计算机能理解的语言。VB的编程语言是相当简单的,它在大部分内容中采用普通英语的词汇和短语。但是,该语言并不模棱两可。当编写Visual Basic语言的语句时,语句绝不会在同一上下文中具有歧义。

新术语 代码是所编写的编程语句的另一种称呼法。

随着学习下一学时,读者将学习越来越多的Visual Basic语言的词汇和语法(语法、标点和拼写规则)。读者将使用VB编程语言在所要创建的应用程序中嵌入指令。所写的全部代码必须一起工作来向计算机发出指令。代码就像是在应用程序内将所有图形、文本和过程连在一起的胶。比如,代码告诉一个查书程序如何成为一个查书程序而不是别的什么程序。程序代码告诉应用程序在很多各种可能的输出和用户动作之下该做些什么。

1.3 Visual Basic的各种版本

Visual Basic 6具有几个版本。在本书的编写时,几乎没人知道Microsoft的Visual Basic的市场战略。Visual Basic至少有两个版本:标准/专业版(Standard/Professional Edition)和企业版(Enterprise Edition)。

本书基本上教的是专业版的内容。专业版提供了更多的工具,包括更多的ActiveX内置控件,更好的Internet编程支持,一个Help文件编译器以及改进的数据库访问工具。大多数专业程序员使用专业版。如果Visual Basic 6.0是“标准版”[也称为学习版(Learning Edition)],那么它提供了最便宜使用Visual Basic的途径以及提供给用户一个完整的开发环境、编程语言以及很多与其他版本相同的工具。一些人仅使用标准版,而不需要别的。你可以把这24学时的教程用于标准版的学习,不过如果可能的话,我推荐您使用专业版。

企业版提供给客户/服务器编程人员用于分布式远程计算机和应用程序的扩展工具。Microsoft在网络化的、分布式的环境方面给企业版的用户加强了VB的性能。

如果读者目前还没有VB编译器,那么推荐购买Sams Publishing出版的资源箱。它叫做Sams Teach Yourself Visual Basic 5 in 21 Days:Complete Compiler Edition(ISBN:0-672-31315-4)以及所带的Visual Basic Learning Edition编译器,你可以使用它并按本书内容进行,以及编写书中所提供的VB应用程序。

技巧 大多数程序员需要的只是标准或专业版。企业版是为那些编写网络、客户/服务器环境应用程序的用户开发的。企业版是一种帮助在特殊客户/服务器环境中的编程人员的增强型版本。

1.4 VB编程过程

当读者想使用Visual Basic时,需要按下面基本步骤去做:

1. 启动Visual Basic。
2. 创建新的应用程序或加载一个已有的应用程序。当创建一个新应用程序时,可以使用Visual Basic的VB Application Wizard来编写程序的初始外壳。我们将在下一学时中学到此内容。
3. 使用Visual Basic所提供的调试工具测试应用程序。调试工具可帮助用户找到并消除那些尽管努力避免,但可能仍然出现的程序错误(也就是Bug)。

新术语 Bug是指程序正确执行前必须改正(消除)的程序错误。

4. 将程序编译成为最终的应用程序。
5. 退出Visual Basic。
6. 将应用程序发布给用户。

用户实际上将在一个编写过程中全部按这些步骤进行。这6步并非是顺序进行的6步,而是在你完成应用程序之前所要经历并返回的阶段。

1.5 启动Visual Basic

可从Windows的开始菜单中启动Visual Basic。Visual Basic开发环境本身一般出现在一个Microsoft Visual Basic 6.0子菜单上,虽然读者的VB可能因安装不同而有不同的称呼。读者将

会看到在子菜单上有附加的程序，但是当选择子菜单中的Visual Basic 6.0时，Visual Basic就加载并出现在屏幕上。

一个与图1-1类似的对话框会在启动Visual Basic之后马上出现。读者所看到实际的对话框可能由于VB 6.0的版本不同而略有不同。

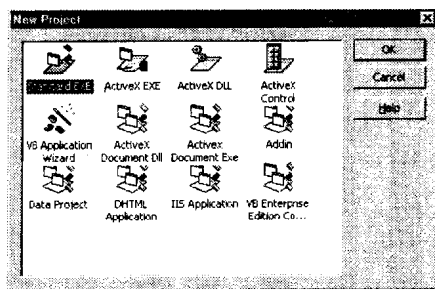


图1-1 当启动VB时，New Project对话框就会出现

注意 总可以通过从VB菜单栏中选择FileNew Project来打开New Project对话框。

当关闭该对话框后，通常就会出现如图1-2所示的Visual Basic屏幕，VB的打开屏幕可能很繁杂！图1-2显示了Visual Basic的开发环境，读者将很快熟悉该环境。读者将在这个开发环境中，创建Windows程序。

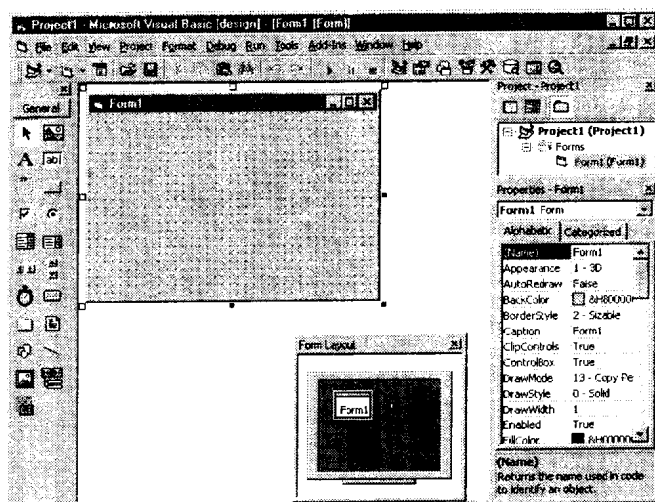


图1-2 开始时VB的屏幕可能显得繁杂

尽管屏幕可能令人迷惑，但是用户完全可以自定义Visual Basic的屏幕来满足自己的需要和喜好。过一会儿之后，读者将可以调整屏幕窗口的大小并隐藏和显示某些窗口，这样Visual Basic屏幕的启动状态会与图1-2的样子大相径庭。

新术语 可停放的窗口(dockable windows)是一个可以重新定大小并移动到屏幕各边，以及可以与其他窗口相连的窗口。

技巧 大多数VB的窗口是可变大小并可停放的，这意味着可以将它们连接在一起、移动它们以及隐藏它们。

在本学时的后面的1.7节“掌握开发环境”中解释了开发环境的各部分以及如何在其中调配。

1.6 退出Visual Basic

从Visual Basic中退出并返回Windows与退出大多数Windows应用程序是一样的：选择FileExit或单击Visual Basic的主窗口的关闭按钮，或按Alt+F4，或者双击屏幕左上角的VB的控制菜单图标。

如果在当前打开的项目中改变了一个或几个文件(记住一个项目是组成应用程序的各文件的集合)，Visual Basic将在退出并返回到Windows之前提示最后一次保存工作。

警告 在没有完全退出Visual Basic之前决不能关闭电源，否则将可能丢失部分或全部当前会话的工作。

1.7 掌握开发环境

在学习Visual Basic之前，学习开发环境的输入与输出，就像是在学开车之前先学习一些汽车的部件知识一样。读者或许想忽略术语学习。但是，如果花时间学习开发环境基本规律的一些内容，将会更好地学习Visual Basic。然后更觉得VB的开发环境很舒服，并能更好地理解本书下面的内容中所涉及的窗口和开发环境中的工具中的有关词条。

图1-3给出了Visual Basic的开发环境，并对其中很多重要的屏幕组件进行了标示。由于可以从菜单和工具条中查看内容，Visual Basic看上去有些像市场上的其他Windows程序。很多Visual Basic的菜单栏命令与其他应用程序(例如Microsoft Word的命令)有相同的运行方式。比如，可以选择Edit|Cut再选择Edit|Paste来从一个位置剪切文本并粘贴到另一个位置。这些同样的菜单命令如今出现在几乎每一个Windows程序中。

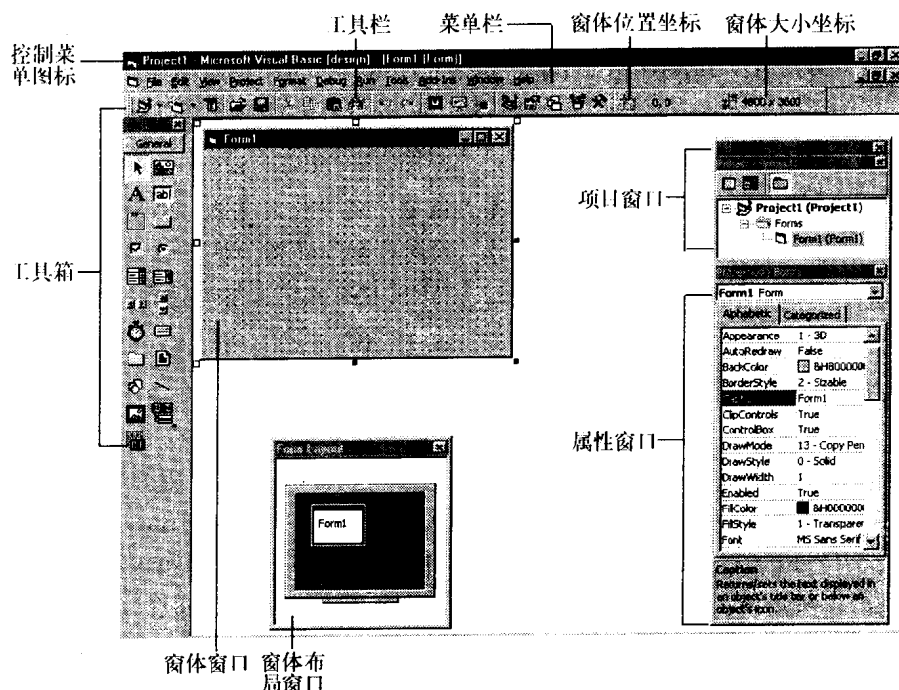


图1-3 了解开发环境

注意 图1-3只显示了开发环境中的一部分窗体和组件。因为将使用附加的工具, 比如 Menu Editor, 本书将讲解如何访问这些工具。

1.7.1 标准组件: 菜单栏和工具栏

Visual Basic的菜单栏和工具栏与其他程序中的作用相同。用户可以单击或按下一个菜单栏选项的热键(比如, Alt+F显示File菜单)来察看一系列下拉菜单选项, 这些菜单选项或提供命令, 或显示另一级菜单或对话框。很多菜单选项具有快捷键(通常称为加速键), 比如Ctrl+S用来打开FileSave选项。当按加速键时, 则不必首先显示菜单再访问选项。

工具栏提供了一个访问很多普通菜单命令的按钮。例如, 不必选择Edit | Paste, 可通过单击Paste菜单栏按钮来实现。和大多数现在的Windows应用程序一样, Visual Basic支持许多工具栏。选择View|Toolbars可查看一系列已有的工具栏。对每一个当前所显示的工具栏, 其名称前会出现选中标记。

技巧 VB采用twips来度量窗体(Form)以及窗体中所包含对象的大小和位置。一个twip是1/1440英寸(可以调节的最小屏幕尺寸)。twip值通常成对出现。第一个位置值表示的是x坐标(twip的值是从屏幕左边开始计算的), 第二个值表示y坐标(从屏幕顶部开始计算的)。0, 0坐标位于屏幕左上角。第一个尺寸值表示了窗体的宽, 第二个尺寸值表明了窗体的高, 因此, 坐标大小1000, 3000指的是当程序运行时Form窗口(窗体窗口)将是1000twip宽, 3000twip高。在下一节中读者将看到Form窗口是编写应用程序的基本窗口。坐标的位置与大小表明应用程序运行时窗体的位置和尺寸。

1.7.2 Form窗口

Form窗口是基本工作区域。虽然Form窗口开始出现时比屏幕的其余部分小, 但是它将组成应用程序的背景。如果用Visual Basic编写一个基于Windows的计算器程序, 那么计算器的按钮全部位于Form窗口上。当运行该计算器时, 所出现的计算器实际上是上面安放有用代码联系起来的组件的应用程序Form窗口。

注意 在Form窗口上将看不见代码。Form窗口具有程序的交互式对象, 比如命令按钮、标签、文本框、滚动条以及其他控件。代码在一个特殊的Code窗口中。在图1-3中并没有Code窗口, 但可以选择View|Code来查看它。Code窗口有点像文本编辑器, 可用它来编写语句, 这些语句集中起来连接应用程序。

注意图1-4窗口中的运行程序的例子。该窗口是一个简单的带有几个选项、文本框和命令按钮的对话框。

图1-4中对话框的编程者先打开一个Form窗口, 再加上一些控件(控件是Form窗口上的组成部分, 起与用户进行交互的作用——有时称为工具), 然后用一些Visual Basic语言代码将各组件联系在一起, 就可以实现上面的功能。这就是编写简单的和复杂的Visual Basic应用程序将要做的内容。用户从一个空白的Form窗口开始, 向其上加入控件, 比如选项按钮和命令按钮。也许, 有些应用程序还要求采用多个Form窗口。

注意 一些应用程序, 比如Word, 允许几个窗体在一个被称为MDI(多文档界面)的模式

中。在这种模式里，用户可以在同一应用程序中打开多个数据文档。一个只要求单独数据窗口的应用程序被称为SDI(单文档界面)应用程序，比如Windows的记事本应用程序，只允许用户一次打开一个数据文档。SDI应用程序或许支持多窗体。但是，除了提供扩展的额外对话框和次级工作屏幕的支持，这些窗体不可具有多个数据文件。

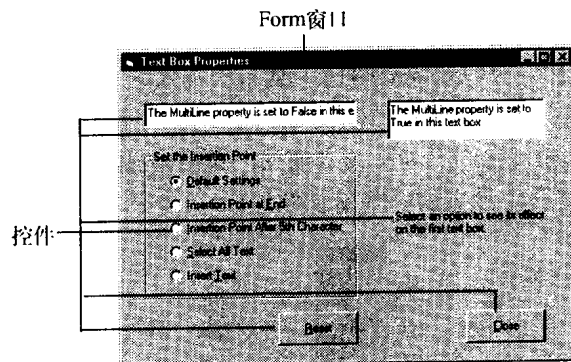


图1-4 一个正在运行的Windows程序的简单对话框

比较图1-4和图1-5。正如所见，图1-5中是在VB开发环境中相同的应用程序的设计状态，而图1-4中是运行状态。设计、创建、编辑、修改应用程序都是在设计阶段进行的。当运行应用程序时，就可以看到工作的结果了。

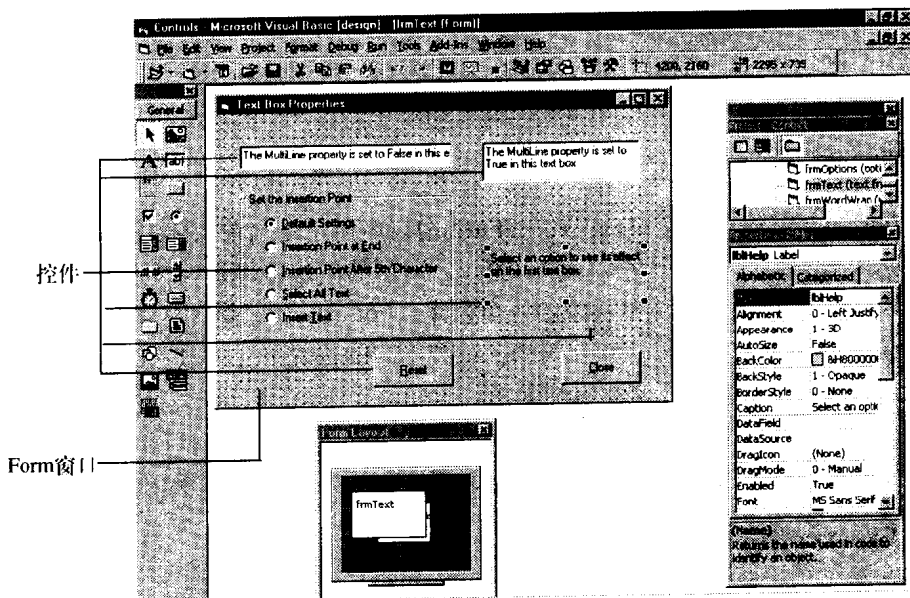


图1-5 在VB开发环境中的对话框

新术语 所创建的应用程序的各部分，例如窗体、代码以及准备输出的图形，组成了源程序。当你或其他用户编辑运行源程序时，VB将该程序翻译成一个可执行程序。用户不可能对可执行程序直接进行修改。如果在运行程序时，发现了错误，则必须将源程序做改动(源程序可能在该项目中包含多个文件)，再重新运行或重新编译源程序。

1.7.3 工具箱提供控件

工具箱包含了可放在Form窗口上的控件。图1-5中所出现的全部控件都在工具箱上。在下一学时中，读者将学习如何将工具箱控件放置在Form窗口上。工具箱上的控件绝不会用完，如果将一个命令按钮放在Form窗口上后，在工具箱中的另一个控件就会做好准备等待你放置。

图1-6给每一个标准工具箱窗口上的工具都标上了名字。这些控件被称为固有控件，因为VB的三个版本都支持这些控件。可以在需要时向工具箱中加入附加的控件。在VB的三个版本中都有一些附加控件，但这些附加控件并不出现在工具箱窗口中，用户可以通过ProjectComponents菜单选项加入它们。如果读者使用的是专业或企业版，那么就可以加入一些在标准版中以及额外控件集中没有的一些附加控件。

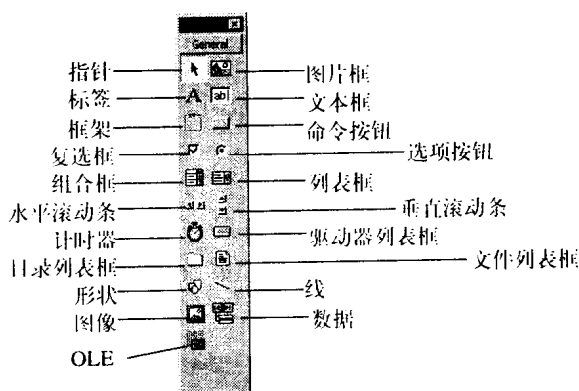


图1-6 VB工具箱所含的固有控件

1.7.4 Form Layout窗口放置窗体

Form Layout窗口(窗体布局窗口)显示了在Form窗口中所示的当前窗体的初始位置与相对大小。比如，回头去看图1-5中的Form Layout窗口。所示的应用程序是一个多窗体应用程序。具有Text Box Properties标题的窗体是其中的一个窗体。Form Layout窗口总显示窗体出现在当前Form窗口的位置，如果需要窗体出现在与当前位置不同的位置上，可以在Form Layout窗口中将窗体移动到运行程序时窗体应出现的位置。

本书基本上没有在各图中显示出Form Layout窗口，为的是给Form窗口和其内容增加更多的空间。用户可以从View菜单中来显示Form Layout窗口，并可以单击其关闭按钮来隐藏它。

1.7.5 Project Explorer窗口

Project Explorer窗口(项目窗口)通常被称为Project窗口，它提供应用程序中全部文件的一个树状结构视图。Microsoft在版本4到版本5中将Project窗口改名为Project Explorer窗口，为的是强调与Windows NT和Windows 95中普遍采用的典型的Explorer树状文件窗口的相似，可以扩展和压缩视图的细节内容。

Project Explorer窗口显示了窗体、模块(应用程序中保存支持代码的文件)、类(高级模块)等等。当想处理一个被装载的应用程序的特殊部分时，可双击Project Explorer窗口中的组件，使其具有焦点。换句话说，如果Project Explorer窗口显示三个窗体，用户需要编辑其中之一

个，可以定位并双击Project窗口中窗体的名字来激活Form窗口中的窗体。图1-7表示了Project Explorer窗口，它包含了几种文件。

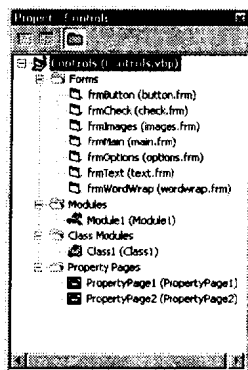


图1-7 Project Explorer窗口包含项目组件

警告 如果将一个帮助文件加到应用程序中，Project窗口并不显示该帮助文件。

1.7.6 Properties窗口

新术语 属性(Properties)是对一个控件详细描述的信息，当每次单击一个不同Form窗口工具后，在Properties窗口中会出现不同的列表。Properties窗口描述了窗体和其控件的属性(外观和功能消息)。很多属性是大多数Visual Basic的每个对象都具有的。Properties窗口列出了Form窗口中选中的控件的全部属性。

1.8 帮助系统

Visual Basic的联机帮助系统是市场上最先进的系统之一。当想得到一个关于控件、窗口、工具或命令的帮助文件，按F1即可。Visual Basic会分析用户正在做什么并提供帮助。此外，Visual Basic支持一个被称为Books Online(联机手册)的大型帮助文件库。当用户选择Help菜单中的Books Online时，Visual Basic会显示出一个关于所能查寻并阅读的关于Visual Basic的树状结构视图。联机帮助还可以与Internet联接。如果读者有一个Internet的联接系统，则可以通过选择HelpMicrosoft on the Web来浏览最新的帮助内容。

1.9 小结

本学时快速地介绍了Visual Basic。也许，读者已经可以看懂比第一次更多的Visual Basic内容了。程序员可以使用Visual Basic来创建极为高级的Windows应用程序，如果读者理解了VB的目的和如何启动和停止它，那么就可准备真正学习它了！

下一学时将描述一个示例的应用程序，以便读者能更好地了解Visual Basic的组件是如何在一起工作的。

1.10 问答

问：我必须学习一种新语言来使用Visual Basic吗？