



轻松学电脑系列教材丛书

中等职业技术学校
电脑学校 **优选教材**

中文 Visual Basic 6.0

实例教程

王彬华 主编



电子科技大学出版社

中文 Visual Basic 6.0 实例教程

主 编：王彬华

副主编：伍云辉 陈 林 刘盛军

编 委：王彬华 潘传伟 李建华 马晓阳 胡宝林

刘小平 苟 杰 伍云辉 刘盛军 谢美泉

石文旭 王 跃 冯秋芳 易冬梅 杜 艳

田 江 刘 庆 冯 欢 陈 林 程 鹏

王建伟 蒲秋实

电子科技大学出版社

内 容 提 要

Visual Basic是美国微软公司开发的面向对象的一种可视化编程语言。

本书由浅入深，系统、全面地介绍了Visual Basic的基本功能和设计思想。着重讲解了可视化语言的概念；Visual Basic的语言基础知识；结构化程序设计知识；面向对象的程序设计概念与方法及编程中常用的算法等。书中每章都用小程序实例来介绍Visual Basic在数据库领域方面的应用，并在每个章节后配有大量的习题，以帮助理解和巩固每章的要点，使读者能够在较短的时间内掌握，并能自如地运用Visual Basic进行程序设计。

本书共有十四章，前面十三章着重讲解了程序设计的一些基础知识。第十四章以程序开发实例为重点，使读者能够充分地理解VB的设计思想，学以致用，达到近快上手的目的。

该书既适用于中等职业技术学校、技工学校、电脑学校和计算机等级考试作为教材，也可作为大专院校及相关院校专业师生的教学参考书和教材，更是广大从事程序设计者必备工具用书。

图书在版编目（CIP）数据

中文 Visual Basic 6.0 实例教程/王彬华主编. —成都：

电子科技大学出版社，2004.1

ISBN 7-81094-402-9

I. 中... II. 王... III. BASIC 语言—程序设计—教材
IV. TP312

中国版本图书馆 CIP 数据核字（2004）第 001550 号

中文 Visual Basic 6.0 实例教程

王彬华 主编

出 版：电子科技大学出版社 （成都建设北路二段四号）
责任编辑：周友谊
发 行：新华书店经销
印 刷：成都市墨池教育印刷总厂
开 本：787mm×1092mm 1/16 印张 15.25 字数 372 千字
版 次：2004 年 1 月第 1 版
印 次：2004 年 1 月第 1 次印刷
书 号：ISBN 7-81094-402-9/TP-238
印 数：1—5000 册
定 价：18.00

■版权所有 侵权必究 敬告读者 勿购盗版■ 本书如有缺页、破损、装订错误，请寄回印刷厂调换。

前 言

当今社会已进入电脑化时代，电脑已广泛用于传统的设计、制造、编辑出版、广告制作等行业，而另一方面专业院校及电脑学校如雨后春笋般涌现，但是，与之相配套的计算机教材却往往不尽人意，普遍缺乏实用性、专业性、可操作性，导致多数电脑专业学校不得不自己编写讲义，所以适合教学的好教材非常缺乏。为此，创世纪计算机工作室在对目前计算机教材使用情况调查和研究的基础上，结合学校的教学实践编写了本套易教易学、轻松有趣的计算机教学丛书。本丛书是我们总结多年教学实践经验的结果。我们希望本丛书不仅为你提供了一套学习的教材，更希望你奉献了一个全新的电脑学习方案：完整的课程安排、丰富的实例讲解、学以致用课后作业，无微不至的设计都是为了达到使你获得最佳的学习效果的目的。

本系列丛书共十本，包括计算机基础、办公应用、程序设计、图形图像及网页制作等方面的内容。

本丛书具有如下特色：

定位准确 明确定位**中等职业技术学校及电脑学校**，丛书坚持基础、技巧、经验并重；理论、操作、提高并举，尤其对初、中级学者使用软件容易出现的疏忽、困惑、难点进行重点突破。

售后服务 可提供售后网上服务；提供后期技术支持；开展网上调查、勘误、答疑、交流、收集反馈信息。读者还可通过电子邮件 wbh@scwbh.com 或 ccwh@163.com 与作者交流。同时，在我们网站 <http://www.scwbh.com> 的论坛中也提供了交流场所，并提供免费下载的汉化软件补丁、程序源代码及实例效果图，下载地址：<http://www.scwbh.com/down> 中“实例教材系列”专栏。图书质量监督电子邮件：lql@scwbh.com。

经过紧张的组织、策划和创作，本丛书已全部出版，承蒙读者厚爱，今后我们还会不断推出实例型的其他姊妹篇。

由于作者水平及时间之原由，书中不足之处望广大读者批评指正。

编 者

2004年1月

目 录

第 1 章 Visual Basic 6.0 概述	1
1.1 什么是 Visual Basic	1
1.1.1 概述	1
1.1.2 Visual Basic 的特点	1
1.1.3 Visual Basic 版本简介	2
1.2 Visual Basic 的集成开发环境	3
1.2.1 启动 Visual Basic	3
1.2.2 Visual Basic 的用户界面	4
1.3 初识 Visual Basic 程序	7
1.3.1 新建工程	8
1.3.2 放置控件	8
1.3.3 设置对象属性	9
1.3.4 编写事件代码	9
1.3.5 调试	11
1.3.6 编译成可执行文件	11
1.4 获得帮助	11
练 习	12
第 2 章 Visual Basic 程序设计基础	13
2.1 Visual Basic 应用程序的结构	13
2.1.1 模块化程序设计	13
2.1.2 基本模块	14
2.1.3 事件驱动应用程序的工作方式	14
2.2 数据类型	15
2.2.1 标识符	15
2.2.2 数据类型	15
2.3 变量	19
2.3.1 变量的命名	19
2.3.2 声明变量	19
2.3.3 变量的作用范围	21
2.3.4 静态变量	22
2.4 常量	24
2.4.1 直接常量	24
2.4.2 符号常量	24
2.4.3 系统常量	25
2.5 表达式与运算符	25
2.5.1 算术表达式	25
2.5.2 比较表达式	26
2.5.3 逻辑表达式	26
2.5.4 字符串连接运算表达式	27

2.6	基本语句	27
2.6.1	编码基础.....	27
2.6.2	赋值语句.....	28
2.6.3	注释语句.....	28
2.6.4	程序暂停和退出语句.....	28
2.6.5	输出语句.....	29
2.6.6	消息对话框(MsgBox)	30
2.6.7	输入框(InputBox)	32
	练习	33
第3章	流程控制	34
3.1	程序结构概述	34
3.2	分支结构	34
3.2.1	If...Then 语句.....	34
3.2.2	If...Then...Else 语句	35
3.2.3	多分支语句 Select Case	37
3.2.4	使用 IIF 函数.....	38
3.3	循环结构	38
3.3.1	Do...Loop 语句.....	39
3.3.2	For...Next 语句.....	42
3.3.3	For Each...Next 语句.....	43
3.3.4	循环嵌套.....	43
	练习	45
第4章	数组	46
4.1	数组的概念	46
4.2	固定数组	47
4.2.1	一维数组	47
4.2.2	多维数组	49
4.3	动态数组	50
4.3.1	创建动态数组.....	50
4.3.2	保留动态数组的内容.....	51
4.3.3	数组的操作函数.....	51
	练习	52
第5章	程序与函数	53
5.1	Sub 程序	53
5.1.1	子过程的定义.....	53
5.1.2	Sub 子过程的分类.....	54
5.1.3	通用过程的创建.....	55
5.1.4	通用过程的调用.....	56
5.2	Function 程序	56
5.2.1	Function 过程的定义.....	57
5.2.2	Function 过程的创建和调用.....	57
5.3	参数传递	58
5.3.1	参数传递方式.....	58
5.3.2	使用可选的参数.....	59
5.3.3	使用不定数量的参数.....	60
5.3.4	用命名的参数创建简单语句.....	60

5.4	常用内部函数	61
5.4.1	数学函数	61
5.4.2	字符串函数	62
5.4.3	转换函数	63
5.4.4	判断函数	63
5.4.5	其他函数	64
练 习		64
第 6 章	面向对象程序设计基础	65
6.1	对象和类概述	65
6.1.1	对象的概念	65
6.1.2	类的概念	65
6.2	Visual Basic 中的对象	66
6.2.1	对象的属性	66
6.2.2	对象的方法	68
6.2.3	对象响应的事件	69
6.3	系统对象	71
6.3.1	屏幕对象(Screen)	71
6.3.2	打印机对象(Printer)	71
6.3.3	剪贴板对象	72
6.4	对象变量和对象数组	73
6.4.1	对象变量	73
6.4.2	对象数组	75
练 习		76
第 7 章	用户界面设计	77
7.1	用户界面概述	77
7.1.1	概述	77
7.1.2	用户界面类型	77
7.1.3	用户接口界面——窗体的设计准则	78
7.2	窗体(Form)	78
7.2.1	创建窗体	78
7.2.2	设置窗体属性	79
7.2.3	窗体的事件	83
7.2.4	窗体的方法	86
7.2.5	应用实例	87
7.3	标签(Label)	89
7.4	文本框(TextBox)	90
7.4.1	文本框的属性	90
7.4.2	文本框的事件	91
7.4.3	应用实例	91
7.5	命令按钮(Command Button)	92
7.5.1	命令按钮的属性	92
7.5.2	命令按钮的事件	93
7.5.3	应用实例	93
7.6	单选按钮(OptionButton)	94
7.7	复选框(CheckBox)	96
7.8	框架(Frame)	97

7.9	定时器控件(Timer)	98
7.10	列表框(ListBox)和组合框(ComboBox)	101
7.10.1	列表框和组合框的常用属性	101
7.10.2	列表框和组合框的常用方法	104
7.10.3	列表框和组合框的常用事件	104
7.10.4	应用实例	104
7.11	滚动条(HScrollBar 和 VScrollBar)	106
7.11.1	滚动条的常用属性	106
7.11.2	滚动条的常用事件	107
7.11.3	应用实例	107
	练习	108
第 8 章	绘图处理	109
8.1	坐标系统和颜色	109
8.1.1	坐标系统概述	109
8.1.2	改变对象的坐标系统	110
8.1.3	使用 Visual Basic 颜色	112
8.2	线条(Line)和形状控件(Shape)	113
8.2.1	线条控件	113
8.2.2	形状控件	113
8.3	图像框控件(Image)	114
8.3.1	图像框的常用属性	115
8.3.2	图片装载函数 LoadPicture	115
8.3.3	应用实例	116
8.4	图片框控件(PictureBox)	117
8.4.1	图片框的绘图方法	118
8.4.2	应用实例	123
	练习	124
第 9 章	高级用户界面设计	125
9.1	添加高级控件	125
9.1.1	添加控件	125
9.1.2	使用集合	126
9.2	图像列表控件(ImageList)	127
9.2.1	图像列表控件的属性和方法	127
9.2.2	ListImage 对象 ListImages 集合	128
9.3	工具栏控件(ToolBar)	129
9.3.1	工具栏控件的常用属性和事件	129
9.3.2	设置工具栏属性	130
9.3.3	为工具栏中各按钮编写代码	131
9.4	树形视图控件(TreeView)	131
9.4.1	TreeView 控件的常用属性和事件	131
9.4.2	在 TreeView 控件中编写代码	132
9.5	列表视图控件(ListView)	134
9.5.1	ListView 常用属性和事件	134
9.5.2	设置 ListView 控件的属性	135
9.5.3	应用实例	137
9.6	状态栏控件(StatusBar)	138

9.6.1	状态栏的常用属性和事件.....	138
9.6.2	设置状态栏属性.....	139
9.7	公用对话框控件(CommonDialog).....	140
9.7.1	“打开”/“另存为”对话框.....	141
9.7.2	“颜色”对话框.....	143
9.7.3	“字体”对话框.....	143
9.7.4	“打印”对话框.....	145
9.7.5	“帮助”对话框.....	146
9.7.6	CommonDialog 控件例子.....	146
练习	148
第 10 章	菜单设计.....	149
10.1	概述.....	149
10.1.1	菜单的组成.....	149
10.1.2	弹出式菜单.....	150
10.1.3	菜单编辑器.....	150
10.2	建立菜单.....	151
10.2.1	使用菜单编辑器中的列表框.....	151
10.2.2	创建菜单的步骤.....	151
10.2.3	通过属性定制菜单.....	152
10.2.4	编写菜单控件的代码.....	154
10.3	弹出式菜单设计.....	155
10.4	应用实例.....	156
练习	158
第 11 章	文件处理.....	159
11.1	文件系统控件.....	159
11.1.1	驱动器列表框(DriverListBox).....	159
11.1.2	目录列表框(DirListBox).....	159
11.1.3	文件列表框(FileListBox).....	161
11.1.4	实例:使用文件系统控件的组合.....	161
11.2	文件对象模型(FSO).....	162
11.2.1	文件对象模型概述.....	162
11.2.2	文件对象模型的组成.....	163
11.3	FSO 对象模型编程.....	164
11.3.1	使用 FSO 对象模型.....	164
11.3.2	使用驱动器和文件夹.....	165
11.3.3	使用文件.....	168
11.4	传统文件操作.....	171
11.4.1	文件访问类型.....	171
11.4.2	使用顺序文件访问.....	171
11.4.3	使用随机文件.....	173
11.4.4	使用二进制文件访问.....	175
练习	176
第 12 章	多文档界面.....	177
12.1	概述.....	177
12.2	创建 MDI 应用程序.....	178
12.2.1	创建 MDI 窗体.....	178

12.2.2	设计时使用 MDI 子窗体	178
12.2.3	使用 MDI 中的窗体	178
12.3	MDI 窗体应用实例	179
12.3.1	建立 MDI 窗体	180
12.3.2	制作菜单	180
12.3.3	编写代码	181
练习	182
第 13 章	数据库处理	183
13.1	建立数据库	183
13.1.1	概述	183
13.1.2	数据库的概念	183
13.1.3	建立数据库	184
13.2	数据访问控件(Data)	186
13.2.1	数据源控件	186
13.2.2	数据绑定控件	187
13.2.3	DAO 对象模型	187
13.2.4	应用实例	188
13.3	ADO 控件	189
13.3.1	ADO 控件的常用属性	189
13.3.2	ADO Data 控件的事件	190
13.3.3	ADO Data 控件应用	191
13.4	ADO 对象编程	193
13.4.1	ADO 对象模型	193
13.4.2	ADO 存取数据	195
13.4.3	建立数据源	195
13.4.4	打开记录集对象	195
13.4.5	使用记录集	197
13.4.6	实用例程	200
练习	202
第 14 章	Visual Basic 编程实例	203
14.1	显示时间和日期程序	203
14.2	改变字体大小和颜色程序	205
14.3	组合列表框、水平滚动条和垂直滚动条	206
14.3.1	组合框	206
14.3.2	滚动条	206
14.3.3	引入控制并设置属性	207
14.3.4	添加程序代码	208
14.4	色彩调节目程序	209
14.5	改变鼠标指针形状程序	210
14.6	输入框和输出框实例	213
14.7	OptionButton 选项按钮与 CheckBox 复选框按钮实例	218
14.8	运动的汽车	222
14.9	菜单设计实例	225
14.10	【VB 应用程序向导】自动生成应用程序	231

第 1 章 Visual Basic 6.0 概述

Microsoft Visual Basic 是在 Windows 环境下的程序开发工具，它提供了开发 Windows 应用程序的最迅速、最简捷的方法。无论是 Windows 应用程序的资深专业开发人员还是初学者，利用 Visual Basic 可方便地实现开发各种实用的应用程序的目的。

1.1 什么是 Visual Basic

1.1.1 概述

1. Basic 语言

首先看看什么是 Basic? Basic 指的是(Beginners All-purpose Symbolic Instruction Code)。Basic 语言自 1964 年诞生至今，已近四十年了。早期的 Basic 语言简单易学，适合初学者学习，但其功能有限，主要用于教学，很难进行大型应用程序的开发。

随着 Windows 操作系统的推出和流行，微软公司推出了 Visual Basic 程序设计语言。其不但继承了原有 Basic 语法的亲和性，还强化了程序结构，拥有多种数据类型。模块化的程序设计，使得程序的开发和维护更为容易，因此非常适用于大型系统的开发。

2. Visual “可视的”

Visual Basic 是在原有的 Basic 语言基础上的进一步发展，它包含了数百条语句、函数及关键词，其中很多与 Windows GUI 有直接的关系。

Visual Basic 可视化的用户界面设计功能，把程序设计人员从繁琐复杂的界面设计中解脱出来；可视化编程环境的“所见即所得”功能，使界面设计如积木游戏一般，编程变成了一种享受；其强大的多媒体功能可轻而易举地开发出集声音、动画、影像和图片于一体的多媒体应用程序；新增的网络功能提供了编写 Internet 程序的能力。

另外，在 Microsoft Office 中的 VBA(Visual Basic for Application)和脚本语言 VBScript(Visual Basic Scripting Edition)都是 Visual Basic 语言的子集，因此学会了 Visual Basic 语言后也可应用到这些领域。

1.1.2 Visual Basic 的特点

Visual Basic 是新型的计算机程序设计语言，与传统程序设计语言相比，Visual Basic 有许多特点，最突出的特点是可视化、事件驱动和交互式。

1. 可视化

在面向过程的程序设计中开发程序时，看不到程序的实际效果，必须在程序运行时才

能看到效果, 如果不满意, 需要回到设计阶段重新设计, 这样的程序调试方式影响了编程的效率。Visual Basic 是可视化程序开发工具, 在开发过程中所看到的界面, 就是程序运行得到的结果。同时, Visual Basic 还为编程人员提供了全面的 Windows 标准控件, 开发人员在设计界面时, 只要将所需控件拖放到窗口的指定位置即可, 整个界面设计过程基本上不需要编写代码, 从而大大提高了系统开发的效率。

2. 事件驱动

在传统的或“过程化”的应用程序中, 应用程序自身控制了执行哪一部分代码和按何种顺序执行代码。执行代码的一般顺序为: 从第一行代码开始并按应用程序中预定的路径执行, 必要时调用过程。

在事件驱动的应用程序中, 代码并不是按照预定的路径执行, 而是在响应不同的事件时执行不同事件所对应的代码片段。事件可以由用户操作触发, 也可以由来自操作系统或其他应用程序的消息触发, 甚至可由应用程序本身的消息触发。这些事件的顺序决定了代码执行的顺序, 因此应用程序每次运行时所经过的代码的路径都有可能不同。

因为事件响应的顺序是无法预测的, 所以在代码中必须对执行时的各种状态做一定的假设。当做出某些假设时(例如, 假设在运行某一输入字段的过程之前, 该输入字段必须包含确定的位), 应该组织好应用程序的结构, 以确保该假设始终有效(例如, 在输入字段中有值之前禁止使用启动该处理过程的命令按钮)。

在执行中代码也可以触发事件。例如, 在程序中改变文本框中的文本将引发文本框的 Change 事件。如果 Change 事件中包含有代码, 则将导致该代码的执行; 如果原来假设该事件仅能由用户的交互操作所触发, 则可能会产生意料之外的结果。正因为这一原因, 所以要求用户在设计应用程序时必须了解事件驱动模型并牢记在心。

3. 交互式

使用传统语言开发应用程序的过程可以分为 3 个步骤: 编码、编译和测试代码。但是 Visual Basic 与传统的语言不同, 它使用交互式方法开发应用程序, 使 3 个步骤之间不再有明显的界限。

在大多数语言里, 如果编写代码时发生了错误, 则在开始编译应用程序时该错误就会被编译器捕获, 此时必须查找并改正该错误, 然后再次进行编译。对每一个发现的错误都要重复这样的过程。而 Visual Basic 是解释性语言, 在编程者输入代码时使进行解释, 即时捕获并突出显示大多数语法或拼写错误, 看起来就像一位专家在监视代码的输入。

除具有即时捕获错误的功能之外, Visual Basic 在输入代码时部分地编译该代码。当准备运行和测试应用程序时, 只需极短的时间即可完成编译。如果编译器发现了错误, 则将错误突出显示于代码中, 这时可以更正错误并继续编译, 而不需从头开始。

正是因 Visual Basic 的这一交互特性, 使得用户在开发应用程序的同时就可以发现自己正频繁地运行着应用程序。通过这种方式, 代码运行的效果在开发时即可进行测试, 而不必等到编译完成以后。

1.1.3 Visual Basic 版本简介

Visual Basic 有三种发行版本, 可以满足不同的开发需求。

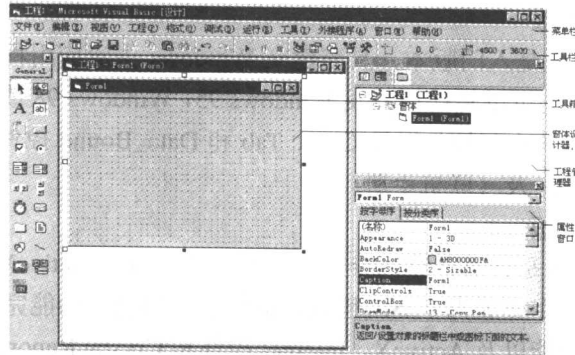


图 1-2 Visual Basic 的用户界面

1.2.2 Visual Basic 的用户界面

如图 1-2 所示就是 Visual Basic 的用户界面。在屏幕上有菜单栏、工具栏、工具箱、窗体设计器、工程管理器、属性窗口等。另外，还有几个隐藏的窗口，分别是窗体布局窗口、调色板窗口、调试窗口、对象浏览器窗口和菜单编辑窗口。下面逐个进行介绍。

1. 标题栏及工作模式

Visual Basic 标题显示当前工程名称以及目前集成开发环境的工作模式，一共有三种工作模式：

(1) 设计模式(Design)

当处于程序开发阶段时，Visual Basic 会在标题栏显示“设计”字样。在此模式下，可进行拖放控件、设置控件属性、编辑事件代码等工作。

(2) 运行模式(Run)

当设计好程序后，按下运行按钮或 F5 键后，程序就开始运行，这时 Visual Basic 进入执行模式，标题栏显示“运行”字样。

(3) 中断模式(Break)

在程序执行过程中，若需中断或进行调试，可按下中断按钮或 Ctrl+Break，Visual Basic 进入中断模式，标题栏显示“中断”字样，立即出现一个窗口，供程序设计或验证用。

2. 菜单栏

菜单栏是 Windows 应用程序中常见的组件。在 Visual Basic 中共有 13 个菜单，可以完成绝大多数的工作。

3. 工具栏

工具栏位于菜单之下，主要作用是将菜单中经常使用的功能以图标的方式提取出来，让用户使用方便，省去选择菜单的麻烦，这也是 Windows 应用程序中常用的组件。

Visual Basic 有多个工具栏，可通过选择菜单【视图】→【工具条】来打开或关闭各工具栏。

注意：工具栏与下面要介绍的工具箱的区别。

4. 工具箱

Visual Basic 的可视化设计功能大部分都是通过工具箱来体现的，工具箱如图 1-3 所示。

工具箱中的每个图标称之为控件(Control)，每个控件都是已经定义好的对象，具有完整的属性、事件和方法，我们只需要学习如何使用，就可编制出各种各样的 Windows 程序。

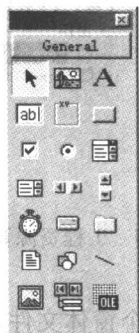


图 1-3 工具箱

如图 1-3 所示的是基本组件，我们可以选择菜单【工程】→【部件】加入其他控件来扩展 Visual Basic 的功能，如图 1-4 所示。有许多第三方厂商开发的控件(其扩展名为.OCX)供 Visual Basic 使用。

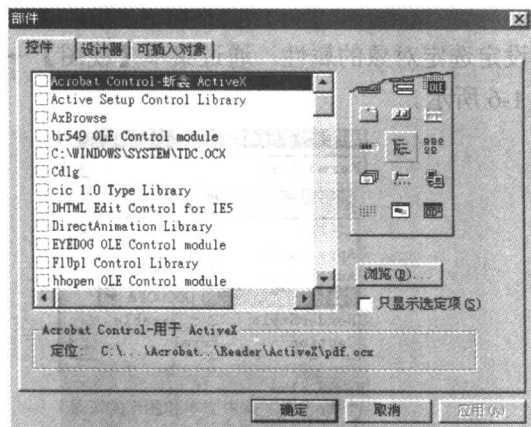


图 1-4 添加控件

5.窗体设计器

窗体(Form)是 Visual Basic 最基本的对象，提供与用户交互的窗口。每当 Visual Basic 启动时，会打开一个窗体，默认名称为 Form1，如果应用程序需要多个窗体，可以选择菜单【工程】→【添加窗体】来创建新窗体。在程序设计阶段所有窗体都是可见的，但在编程中设置属性为隐藏窗体，这样在程序运行时，所隐藏的窗体将不可见。

将工具箱中的控件拖放到窗体中，就可建立用户交互的对话框。

6.工程资源管理器

工程资源管理器主要用于管理工程。每个用 Visual Basic 设计的应用程序，在设计时被称为一个工程(Project)。每个工程可能会使用许多不同的文件，这些文件间的相互关系由工程资源管理器来记录、管理，如图 1-5 所示。



图 1-5 工程资源管理器

常用文件类型简单介绍如下:

(1)窗体文件(.Frm)每个窗体对应一个窗体文件,用来保存窗体的设计(控件有其属性、程序代码等)。

(2)模块文件(.Bas)用来保存一般的程序模块。当选择菜单【工程】→【添加模块】时,便建立了一个扩展名为.Bas 的模块文件。

(3)工程文件(.Vbp)保存整个工程中所使用到文件的路径及文件名。

7.属性窗口

属性窗口是用于设定选定对象的属性。通过菜单【视图】→【属性窗口】或按 F4 可打开属性窗口,如图 1-6 所示。



图 1-6 属性窗口

属性窗口包括对象选择栏和属性表两部分。

(1)对象选择栏

位于属性窗口的上方,这一栏的右侧有一个倒三角形,按一下便会显示出对象列表,包括窗体及其中的控件。我们可以在对象列表中选择对象,该对象的属性会显示在下方的属性表中。

(2)属性表

属性表的内容随选择的对象而变化,属性的排列顺序以英文字母排列,也可以使用分类排序。

属性表分为左右两栏,左边是属性名称,右边是属性值。设置属性值时可以使用以下三种方法:

- ①通过键盘直接输入属性值;
- ②单击属性值右边的下拉箭头,在列出的选项中选择;
- ③单击属性值右边的选择按钮,打开一个对话框来设置属性。

8. 窗体布局窗口

在窗体布局窗口中可以使用表示屏幕的小图像来布置应用程序中各窗体的位置，如图 1-7 所示。

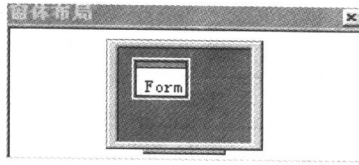


图 1-7 窗体布局

9. 代码编辑窗口

在窗体或窗体上的控件上双击鼠标左键，便会弹出如图 1-8 所示的代码编辑窗口。

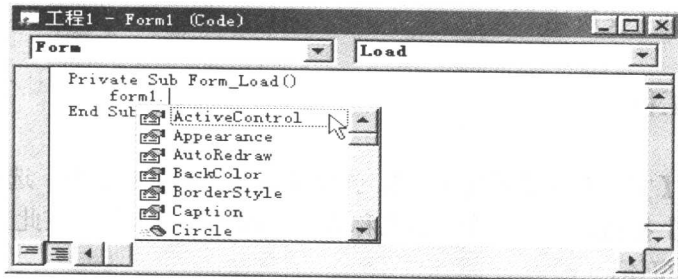


图 1-8 代码编辑窗口

代码编辑窗口是输入应用程序代码的编辑器，应用程序的每个窗体或代码模块都有一个单独的代码编辑窗口。代码编辑窗口由三部分组成：

(1) 【对象】下拉列表框

在这个列表框中包含了当前窗体中所有的对象。通过此下拉列表框选择相应的对象，就可针对相应的对象编写代码。

(2) 【事件】下拉列表框

这里列出了左边所选择的对象能响应的事件名称。

(3) 代码编辑区

代码编辑区是程序代码输入和编辑的区域。它有两种显示方式：过程查看和全模块查看，可以通过代码区左下角的两个按钮切换。其中过程查看只显示指定对象指定事件的代码，而全模块查看则显示了当前窗体每个对象能响应的所有事件的代码。

Visual Basic 的代码编辑器最有特色的是其代码自动补全特性，如自补全语句、属性、事件、方法、函数单词以及语法等。如图 1-8 所示，当我们输入对象 Form1，然后再输入小数点，就会弹出一个包含 Form1 的属性、方法的列表框供我们选择。

以上介绍了开发环境中的常用组件，其他窗口将在使用过程中介绍。

1.3 初识 Visual Basic 程序

通过 Visual Basic 建立应用程序非常简单，下面我们来设计一个简单的欢迎程序，此程序执行结果如图 1-9 所示，其功能如下：

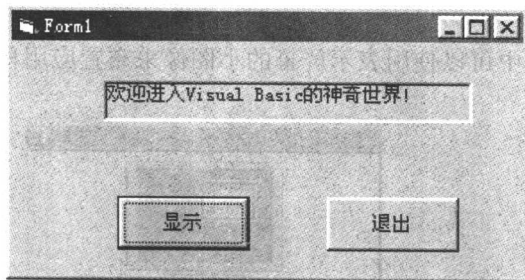


图 1-9 程序运行结果

- (1)当用户单击【显示】按钮时，在窗体上显示“欢迎进入 Visual Basic 的神奇世界!”;
 - (2)单击【退出】按钮退出程序。
- 具体步骤如下文所述。

1.3.1 新建工程

选择菜单【文件】→【新建工程】，弹出【新建工程】对话框，选中【标准 EXE】，然后单击【确定】按钮，建立一个缺省名称为“工程 1”的工程，在此工程中会有一个缺省名为 Form1 的窗体。

1.3.2 放置控件

根据程序需要，将所需控件放在窗体的适当位置。该例中需要三个控件，一个标签控件(Label)用来显示文字，两个命令按钮(Command Button)用来接收用户的操作。

首先我们来学习如何在窗体上放置控件。

1.添加控件:

方法一:

- (1)双击工具箱中要使用的控件，该控件就会出现在窗体的正中央;
- (2)将控件移至适当位置并调整其大小。

方法二:

- (1)用鼠标左键单击所需控件;
- (2)移动鼠标到窗体上适当的位置，此时光标为十字形，然后拖动鼠标，即可以在窗体上画出控件。

2.删除控件

- (1)在窗体上单击要删除的控件(该控件周围会出现八个点表示被选定)
- (2)按键盘上的 Del 键即可删除选定的控件。

3.改变控件大小

- (1)在窗体上选取控件;
- (2)将鼠标移至黑点处拖动，即可改变控件大小。

4.移动控件

- (1)在窗体上选中控件;