

看程序实例学

Visual Basic for Windows

(入门篇)

适用于 2.0~3.X 版



鼠标运动会程序
飞舞的蝴蝶程序
小时钟程序
小算盘程序
文书编辑程序
绘图程序

洪
锦
魁

编
著

学苑出版社

计算机语言技术系列丛书

看程序实例学
Visual Basic for Windows
(入门篇)

洪锦魁	编著
吴红艳	改编
朱培华 宝 庆	审校

学苑出版社

1994.

(京)新登字 151 号

内 容 提 要

Visual Basic for Windows 是 Windows 软件开发最强有力的程序设计语言,国内外应用人员日益增多。但因为它的语法规则较多,给初级者学习与使用带来一定的障碍。鉴于此,本书以大量程序实例,逐步解释其基本概念,其中的中文程序,对国内的用户尤有帮助。本书适用于广大 Windows 应用人员自学、提高程序设计能力。

欲购本书的用户,请直接与北京 8721 信箱联系,邮编:100080,电话:2562329。

版 权 声 明

本书繁体字中文版原名为《看程式实例学 Visual Basic for Windows(入门篇)》,由松岗电脑图书资料股份有限公司出版。版权归松岗公司所有。

本书简体字中文版由松岗公司授予北京希望电脑公司和学苑出版社独家出版、发行。未经出版者书面许可,本书的任何部分均不得以任何形式或任何手段复制或传播。

计算机语言技术系列丛书

看程序实例学 Visual Basic for Windows

编 著: 洪锦魁
改 编: 吴红艳
审 校: 朱培华 宝 庆
责任编辑: 甄国宪
出版发行: 学苑出版社 邮政编码: 100036
社 址: 北京市海淀区万寿路西街 11 号
印 刷: 双青印刷厂
开 本: 787×1092 1/16
印 张: 31.875 字数: 740 千字
印 数: 1~5000 册
版 次: 1994 年 1 月北京第 1 版第 1 次
ISBN7-5077-0776-8/TP·8
本册定价: 55.00 元

学苑图书印、装错误可随时退换

用户请注意

欲购本书配套软盘的朋友,请按下列方法汇款:

单价:50.00 元(含邮费)

注:从银行电汇款的朋友请按下列帐号和收款单位汇款:

收款单位:北京希望电脑公司	
开户银行及帐号:北京海淀工商行中关村城市信用社	帐号:05079-08
注:要增值税发票的朋友,请仔细填下表。	

购 货 单 位	名称	纳税人登记号
	地址电话	开户银行及帐号

注:本次共订盘 张 应收款为¥(小写):

注:用户填好此单后请连同此单、信汇单一并传真 01-2561057 或 01-2579874,收到传真后即发货。或邮寄 100080 北京海淀 8721 信箱资料部朱红收
联系电话:01-2562329,2541992 传真:01-2561056,2579874

序

编写本书的目的只有一个,那就是希望读者可以花费最少的时间而学会使用 Visual Basic for Windows 编写 Windows 应用程序。经过笔者多年的教学和自我进修的经验得知,欲达到上述目的,一定要在介绍程序的语法规则时,尽量辅以程序实例作解说。因此,程序实例丰富乃是本书的一大特色。

Visual Basic for Windows 本身的语法规则及命令语句非常多,事实上不可能在一本书中完全解说 Visual Basic 的所有概念。几经考虑后,笔者决定将 Visual Basic 中最常用且最基础的部分,以丰富的程序实例作解说,相信各位读完本书后,将具有设计 Windows 应用程序的入门知识。

Visual Basic for Windows 软件可以在中文或英文 Windows 下执行,由于国内绝大多数的使用者都是使用中文 Windows,因此本书的程序实例也完全以在中文 Windows 下执行作解说。事实上只要你读完本书,相信以后在使用中文 Windows 时,将不再感觉 Windows 应用程序是一个很复杂且难设计的程序。

感谢松岗翁美珠小姐协助笔者购买此软件,感谢松岗杨嘉水处长提供本产品的资料,同时也感谢松岗黄兆铭处长的督促与鼓励方可完成此著作。

编著本书虽力求完美,但水平不足谬误难免,尚祈读者不吝指正。

洪锦魁

1993. 11. 14 于魁工作室

目 录

第一章 Visual Basic 基本概念	(1)
1.1 不一样的工作环境	(1)
1.3 Visual Basic 的特色	(2)
1.4 硬件需求	(3)
1.5 启动 Visual Basic for Windows	(3)
1.6 Visual Basic 环境说明	(4)
1.7 Visual Basic 的工作模式	(9)
1.8 使用工具箱	(10)
1.9 控件的编辑	(13)
1.10 度量单位	(17)
1.11 退出 Visual Basic 环境	(18)
第二章 基本数据处理的概念	(20)
2.1 Debug Window	(20)
2.2 Print 的基本用法	(21)
2.3 变量	(22)
2.4 再谈 Print	(23)
2.5 基本数据类型	(23)
2.6 强制变量类型的声明 Dim	(26)
2.7 算术表达式	(27)
2.8 Visual Basic 特殊运算符	(29)
第三章 简单的程序实例	(30)
3.1 第一个 Windows 应用程序	(30)
3.1.1 建立第一个 Windows 应用程序	(30)
3.1.2 执行所建的 Windows 应用程序	(35)
3.1.3 结束执行 Windows 应用程序	(36)
3.1.4 更改窗体名称	(37)
3.1.5 存储所建的程序文件	(38)
3.1.6 环境分析	(40)
3.2 常用术语说明	(40)
3.3 Visual Basic 几个常用功能介绍	(41)
3.3.1 New Project	(41)
3.3.2 Open Project	(41)
3.3.3 建立可执行文件	(43)
3.4 初值的设定	(44)
3.5 存文件与打印文件的技巧	(49)
3.6 包含中文数据的程序设计	(51)

3.7	注解程序的技巧	(53)
3.8	AUTOLOAD.MAK	(55)
3.9	习题	(56)
第四章	窗体的应用程序设计	(57)
4.1	检测键盘的按键	(57)
4.2	结束程序的执行	(57)
4.3	连接两次鼠标键	(59)
4.4	Paint 事件	(61)
4.5	重绘窗口内容的属性	(65)
4.6	绘图的基本使用原则	(65)
4.7	Resize 事件	(68)
4.8	检测左右鼠标按键	(70)
4.9	绘图颜色的设定	(73)
4.10	MouseMove 事件	(77)
4.11	当前绘图点	(80)
4.12	绘图宽度的控制	(82)
4.13	用打印机输出图形	(84)
4.14	读取输入数据	(91)
4.15	对话框的输出	(96)
4.16	子程序或函数的设计	(102)
4.17	窗体窗口的属性	(105)
4.18	习题	(117)
第五章	功能钮的设计	(119)
5.1	功能钮的建立	(119)
5.2	基本程序设计原则	(121)
5.3	对象名称的更改	(123)
5.4	中文名称的功能钮	(126)
5.5	快速键与功能钮	(128)
5.6	窗口含多个功能钮	(132)
5.7	Disable 功能钮	(140)
5.8	功能钮名称的修改	(145)
5.9	拷贝功能钮	(147)
5.10	功能钮的属性	(148)
5.11	习题	(151)
第六章	标签和正文框	(153)
6.1	标签的建立	(153)
6.2	正文框的建立	(155)
6.3	正文框数据的输入	(157)
6.4	标签与正文框	(159)

6.5	快速键的应用	(162)
6.6	密码的应用	(164)
6.7	取消正文框的字符	(167)
6.8	正文框的 Change 事件	(168)
6.9	综合实例	(170)
6.10	标签的属性	(186)
6.11	正文框的属性	(188)
6.12	习题	(191)
第七章	复选框	(192)
7.1	建立复选框	(192)
7.2	复选框的属性	(198)
7.3	习题	(200)
第八章	选项钮	(201)
8.1	建立选项钮	(201)
8.2	选项钮的属性	(210)
8.3	习题	(212)
第九章	框架	(213)
9.1	建立框架	(213)
9.2	多组框架的应用	(217)
9.3	框架的属性	(225)
9.4	习题	(226)
第十章	列表框	(227)
10.1	建立列表框	(227)
10.2	建立列表数据	(227)
10.3	清除所有列表框数据	(228)
10.4	选取列表框项目	(232)
10.5	删除某个列表项目	(235)
10.6	列表框的属性	(238)
10.7	习题	(240)
第十一章	组合框	(241)
11.1	建立组合框	(241)
11.2	建立列表数据	(242)
11.3	组合框的数据运作	(249)
11.4	组合框的 Style 属性	(250)
11.5	组合框的属性	(255)
11.6	习题	(257)
第十二章	滚动条	(258)
12.1	建立滚动条	(258)
12.2	滚动条的重要属性	(259)

12.3	Scroll 事件	(265)
12.4	滚动条的属性	(266)
12.5	习题	(268)
第十三章	定时装置	(269)
13.1	建立定时装置	(269)
13.2	定时时间的设定	(269)
13.3	数字时钟的设计	(275)
13.3.1	Time \$	(275)
13.3.2	NOW	(278)
13.3.3	Second(Now)	(279)
13.3.4	Minute(Now)	(279)
13.3.5	Hour(Now)	(279)
13.3.6	重新设定坐标轴	(279)
13.4	定时装置的属性	(285)
13.5	习题	(285)
第十四章	菜单的设计	(287)
14.1	建立基本的菜单	(287)
14.2	建立菜单内的命令	(290)
14.3	再谈 Menu Design Window 的功能	(295)
14.4	建立命令内的命令	(297)
14.5	快速键的设定	(303)
14.6	习题	(309)
第十五章	控件数组	(310)
15.1	功能钮的应用	(310)
15.2	菜单的应用	(319)
15.3	习题	(323)
第十六章	图片框	(324)
16.1	图片框的建立	(324)
16.2	将图形载入图片框	(325)
16.2.1	Design Time	(325)
16.2.2	Run Time	(329)
16.3	多个图片框的应用	(330)
16.4	在图片框内绘图	(337)
16.5	图片框的属性	(340)
16.6	习题	(343)
第十七章	影像工具	(345)
17.1	建立影像框	(345)
17.2	Stretch 属性	(346)
17.3	影像框与图片框	(349)

17.4 按图标钮的应用.....	(355)
17.5 影像工具的属性.....	(357)
17.6 习题.....	(358)
第十八章 控件的移动	(360)
18.1 Left 和 Top 属性	(360)
18.2 Move	(368)
18.3 拖动对象的技术 1	(377)
18.4 拖动对象的技术 2	(380)
18.5 习题.....	(382)
第十九章 直线和外形工具	(384)
19.1 建立直线对象.....	(384)
19.1.1 线条的宽度 BorderWidth	(385)
19.1.2 线条形式 BorderStyle	(385)
19.1.3 线条端点的座标.....	(385)
19.1.4 再谈线条工具的目的.....	(386)
19.2 线条工具的属性.....	(388)
19.3 建立外形对象.....	(388)
19.3.1 外形的种类 Shape 属性.....	(389)
19.3.2 外形线条的种类 BorderStyle	(390)
19.3.3 填充外形内部 FillStyle	(390)
19.3.4 外形工具的目的.....	(391)
19.4 外形工具的属性.....	(395)
19.5 习题.....	(396)
第二十章 建立自己的对话框	(397)
20.1 系统内建的对话框.....	(398)
20.2 对话框的格式.....	(398)
20.3 对话框与多重窗口.....	(398)
20.4 切换显示对话框窗口.....	(398)
20.5 正式建立对话框窗口.....	(399)
20.6 独立程序文件.....	(405)
20.7 习题.....	(413)
第二十一章 文件的输入与输出	(414)
21.1 顺序存取.....	(414)
21.1.1 Open	(414)
21.1.2 Close	(415)
21.1.3 适用顺序文件的输入与输出.....	(415)
21.1.4 EOF	(415)
21.2 随机存取.....	(419)
21.2.1 Put	(420)

21.2.2	Get	(420)
21.3	习题.....	(427)
第二十二章	文件、目录和磁盘机列表框	(428)
22.1	磁盘列表框.....	(428)
22.1.1	建立磁盘机列表框.....	(428)
22.1.2	磁盘机列表框的属性.....	(431)
22.2	目录列表框.....	(433)
22.2.1	建立目录列表框.....	(433)
22.2.2	目录列表框的属性.....	(439)
22.3	文件列表框.....	(441)
22.3.1	建立文件列表框.....	(441)
22.3.2	文件的属性.....	(444)
22.3.3	列出所选的文件技术.....	(448)
22.3.4	文件列表框的属性.....	(451)
22.4	习题.....	(453)
第二十三章	区段正文处理技巧.....	(454)
23.1	与区段正文有关的属性介绍.....	(455)
23.2	剪贴簿控件.....	(458)
23.3	习题.....	(460)
第二十四章	通用对话框.....	(461)
24.1	建立通用对话框.....	(461)
24.2	对话框的实例.....	(462)
24.2.1	Open 对话框	(462)
24.2.2	Save As 对话框	(465)
24.2.3	Color 对话框	(467)
24.2.4	Font 对话框	(469)
24.2.5	Print 对话框	(472)
24.3	通用对话框的属性.....	(473)
24.4	习题.....	(475)
第二十五章	声音的设计.....	(477)
25.1	声音产生的技巧.....	(477)
25.2	习题.....	(484)
附录 A	安装 Visual Basic for Windows	(485)
附录 B	IBM PC ASCII 码	(490)
附录 C	本书程序介绍	(498)

第一章 Visual Basic 基本概念

本章是一个入门章节,主要是介绍操作 Visual Basic 所需具备的基本条件,同时也介绍启动与退出 Visual Basic 的方法。读者读完本章,相信必定可以了解下列基本知识:

1. 启动及退出 Visual Basic 的方法。
2. 认识 Visual Basic 的环境。
3. 了解工具箱的使用方法。
4. 了解在 DOS 环境及在 Windows 环境下建立应用程序的差别。
5. 了解 Visual Basic 的度量单位点素(Twip)。

自从 1990 年 5 月 22 日 Microsoft 公司在美国一场壮观的 Microsoft Windows 3.0 发表会后,几乎所有的 PC 软件产品开发即以适用 Windows 环境为目标。轻敲按钮,弹指之间即可协助用户完成工作,的确带给了用户很大的便利。

Windows 环境尽管好用,不过如何设计可以在 Windows 环境下执行的 Windows 应用程序,却一直是程序设计人员的恶梦。基本上各位可以使用下列三种方式设计有关的 Windows 应用程序:

1. 使用 C 语言,配合 Windows API(Application Programming Interface)函数,例如,Visual C++ 或 Borland C++ 均提供以这种方式设计 Windows 应用程序。

2. 使用 C++ 配合 MFC,所谓的 MFC 是 Microsoft Foundation Class Library 的缩写,这是一个专门提供设计 Windows 应用程序的类型函数库,在 C++ 语言中,配合此类型函数库的调用,可以很方便地建立 Windows 应用程序,例如,Visual C++ 提供这种方式设计 Windows 应用程序。

3. 使用 Visual Basic for Windows 软件,这也是本书的主题。事实上,这就是说,可以很容易地使用该软件设计 Windows 应用程序。

1.1 不一样的工作环境

在 DOS 环境下设计 DOS 应用程序时,程序设计人员可以独占性地使用系统资源(例如 CPU、内存、屏幕、键盘等等),执行数据的输入/输出和一般运算。不过在 Windows 环境下,则并不如此单纯,下面将介绍 Windows 应用程序和 DOS 应用程序的工作环境的差异。

1. 多任务(Multitasking)的特性

Windows 是个多任务(Multitasking)的作业环境,也就是在同一时间可能会有好几个应用在执行,在此情况下,所设计的应用程序无法独占性地使用所有系统资源(系统资源是指 CPU、内存、键盘等等)。而 DOS 则只是一个单用户环境,程序设计时,你可以充分使用系统可用资源(available resource)。

例如,对 DOS 应用程序而言,数据一定可在屏幕上执行输出,不可能有其它程序共同分享屏幕资源。但是在 Windows 环境下,一个屏幕可以同时出现好几个应用程序的输出窗口,因此,对于一个 Windows 应用程序而言,在数据输出时,必须考虑与其它正在执行的应用程

序共同分享屏幕资源。所以,所有的 Windows 应用程序必须遵照 Windows 操作系统特有的界面(Interface)执行运算,以确保 Windows 操作系统能有效地管理系统资源,供应用程序使用。

2. 操作界面

在 DOS 环境下,程序设计人员可以独占性使用屏幕,只要执行几个命令(Write 或 Print)调用,即可将数据输出,或是接收所输入的数据。

在 Windows 环境下,若想执行输入或输出动作,首先必须在屏幕上设计一个窗口,然后再通过此视图执行输入与输出。当然同一个应用程序是可能存在多个应用程序窗口的,这样,可形成多文档界面(multiple Document Interface)。

3. 数据的输入

在 DOS 环境下,若有输入动作发生,所输入的数据一定是供当前的正在执行的应用程序使用。在 Windows 环境下,由于有多个应用程序在执行,因此,究竟数据输入时,是由哪一个应用程序接收,将由 Windows 操作系统管理。

4. 数据输出模式

在 DOS 环境下,操作模式是假设在文本(Text)模式下工作,若是想执行图形输出,必须先切换至图形(Graphics)模式,若不想显示图形输出,必须切换回文本模式。在 Windows 环境下,图形模式是基本工作模式,你可以很顺利地使用此模式执行文本、图形数据的输出。

看了以上的叙述可知 Windows 应用程序的设计的确较为困难,事实上以 C 语言设计 Windows 应用程序的确相当复杂,不过用 Visual Basic 设计 Windows 应用程序则不太困难。过去在 DOS 环境下,各位可能知道,汇编语言算是低级语言,以汇编语言设计 DOS 应用程序相当复杂。而 Basic 或 C 则称之为高级语言,它们设计 DOS 应用程序则简单多了。

若以 Windows 的观点来看,则我们可以将以 C 设计 Windows 应用程序想像成低级语言,较复杂且难学。而以 Visual Basic 设计 Windows 应用程序则是高级语言,比较简单好学。

1.3 Visual Basic 的特色

Visual Basic 是 1991 年推出的,用它设计 Windows 应用程序较简单的原因是,一般 Windows 复杂的界面处理动作,例如,滚动条控制(scroll bar)、菜单建立(menu)、对话框(dialog box)……等复杂的动作,皆已被隐藏,读者可用几个步骤轻松地完成工作。

Visual Basic 是一种事件驱动(Event-Driven)的程序概念。你必须在程序内,设计各种事件(event)的处理方法(相当于程序码)。当此事件发生时,随即执行该事件的程序码。例如,你可以在程序内设计下列事件的处理方法。

1. 按关闭钮,可结束应用程序执行。
2. 拖动鼠标,可随着鼠标移动而画线。
3. 按清除钮,可清除窗口内容。

则你所设计的应用程序在执行时,只要有上述 3 个事件的任一事件发生,程序就执行所指定的任务,若是发生了未定义的事件,本应用程序则不予理会。

用 C 语言设计的 Windows 应用程序冗长,但在冗长的程序码中,大约 80%~90%的程序码是用于建立用户界面,而只有约 10%~20%左右的程序码是真正在处理应用程序本

身。Visual Basic for Windows 软件的诞生真是 Windows 应用程序设计人员的一个福音,因为,有了此套软件后,你可以很轻松地使用 Visual Basic 所提供的工具建立用户界面,从而将精力花费在应用程序本身,如此将可以大大地提高设计程序的工作效率。

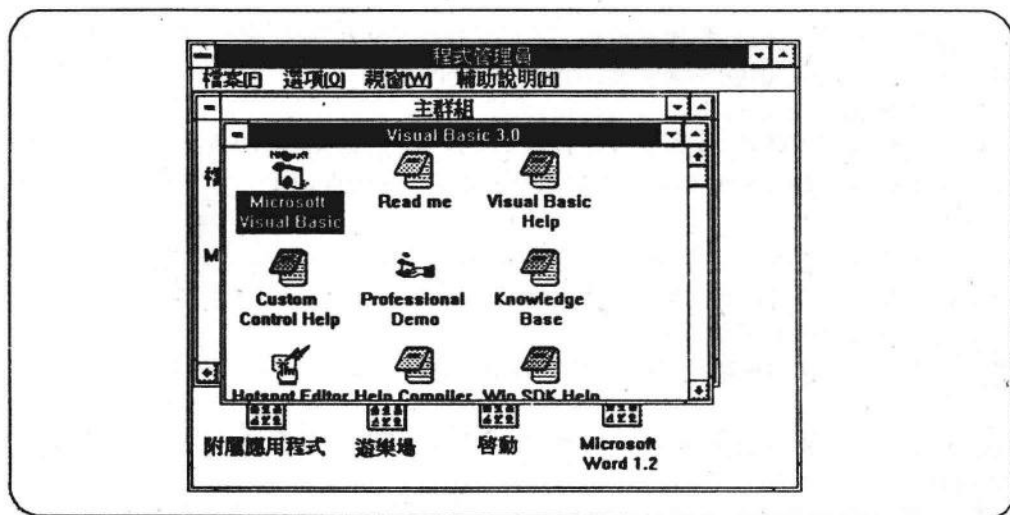
1.4 硬件需求

有关安装 Visual Basic for Windows 的步骤可参考附录 A,至于安装此软件所需的硬件配置则如下所示:

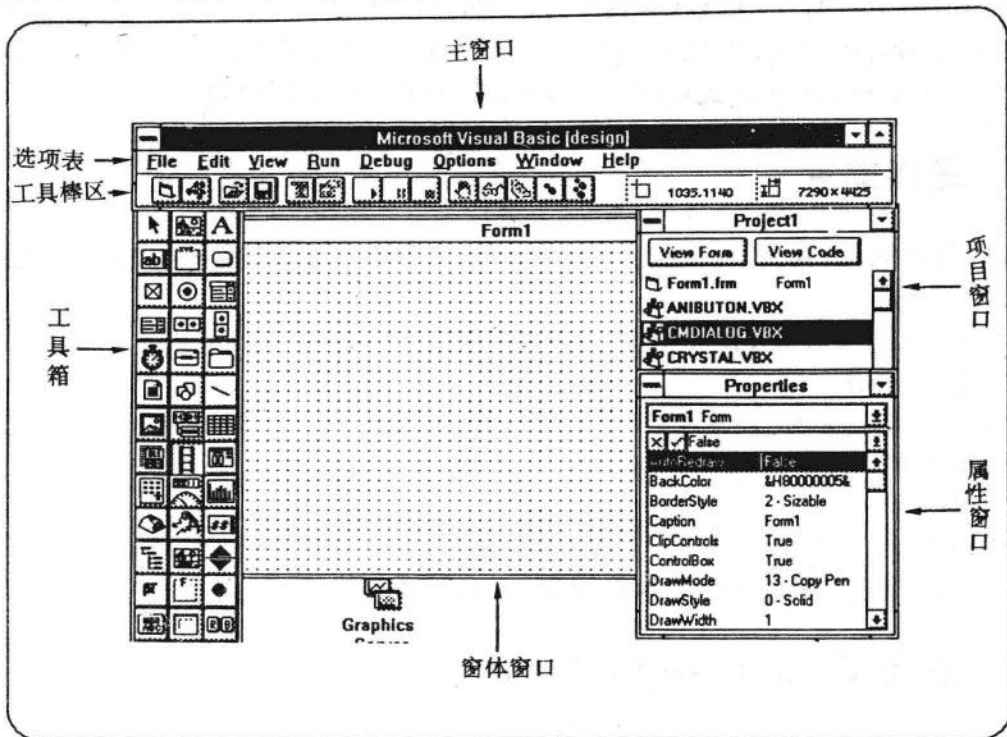
1. IBM PC 或兼容机,同时具有 80286 或更高档的微处理机。
2. 硬盘驱动器。
3. 5¼寸或 3½寸的软盘驱动器。
4. 1M 内存(RAM)。
5. 鼠标。
6. MS-DOS 3.1。
7. 中文版或英文版的 Microsoft Windows 3.0 或更新版。

1.5 启动 Visual Basic for Windows

你可以将 Visual Basic 安装在中文版 Windows 下,也可以将 Visual Basic 安装在英文版 Windows 下,两者间最大差别在于,若将 Visual Basic 安装在中文 Windows 下,以后将可处理中文数据。由于国内的电脑大多安装有中文版的 Windows,因此,本书将以在中文版的 Windows 下讲解此软件。不过本书也适合于在英文版的 Windows 下,使用 Visual Basic 的读者首先请进入中文 Windows 且打开 Visual Basic 群组,如下所示:



连接两次 Microsoft Visual Basic 项目或其图标将可进入 Visual Basic 集成环境,如下所示:



上图是 Visual Basic 3.0 专业版(Professional)的窗口界面,如果你是使用 Visual Basic 3.0 标准版(standard),则在工具箱内只能见到 21 个工具(将在下一节说明)。

1.6 Visual Basic 环境说明

成功地进入 Visual Basic 集成环境后,各位可见到 Visual Basic 窗口(可参考前一小节,一般可以将 Visual Basic 集成环境分成 5 个窗口,如下所示:

1. 主窗口 Main Windows

主窗口内包含一个菜单区(内含 8 个选择项)和一个工具棒区(Toolbar),工具棒区的功能钮主要是提供程序设计时常用的功能,这样各位可以在此直接按一个钮即可执行某个功能。

此外,只要按一下菜单区的某个选择项,即可打开此选择项底下的下拉式菜单(pull down menu),然后各位可按自己的需要执行所要的功能。

下面是工具棒区各钮的意义:

 : 建立一个新的窗体(form)。你也可以执行 File 菜单的 New Form 命令,执行此钮的功能。

 : 建立一个新的模块(module)。你也可以执行 File 菜单的 New Module 命令,执行此钮的功能。

 : 打开现存的项目文件(project file)。你也可以执行 File 菜单的 Open Project 命令,执行此钮的功能。

 : 存储当前的项目文件(project file)。你也可以执行 File 菜单的 Save Project 命令,执行此钮的功能。

 : 显示可设计菜单(menu design)的窗口,以设计某个应用程序的菜单。你也可以

执行 Windows 菜单的 Menu Design 命令,执行此钮的功能。

 : 显示属性窗口(Properties Window),若属性窗口被遮住而不在屏幕上显示,则可以按此钮,以便显示它。也可以执行 Windows 菜单的 Properties 指令,执行此钮的功能。

 : 正式执行当前所设计的 Windows 应用程序。你也可以执行 Run 菜单的 Start 命令,执行此钮的功能。

 : 当 Windows 应用程序在执行时,令程序暂停执行。你也可以执行 Run 菜单的 Break 命令,执行此钮的功能。

 : 当 Windows 应用程序在执行时,令程序停止执行,同时返回 Visual Basic 屏幕,以便可以继续编辑或设计程序。你也可以执行 Run 菜单的 End 命令,执行此钮的功能。

 : 在当前执行行内切换中断点,通常是用于调试(debug)程序阶段。你也可以执行 Debug 菜单的 Toggle Break 命令,执行此钮的功能。

 : 在编码窗口(Code Window)内,显示当前选项的值。你也可以执行 Debug 菜单的 Instant Watch 命令,执行此钮的功能。

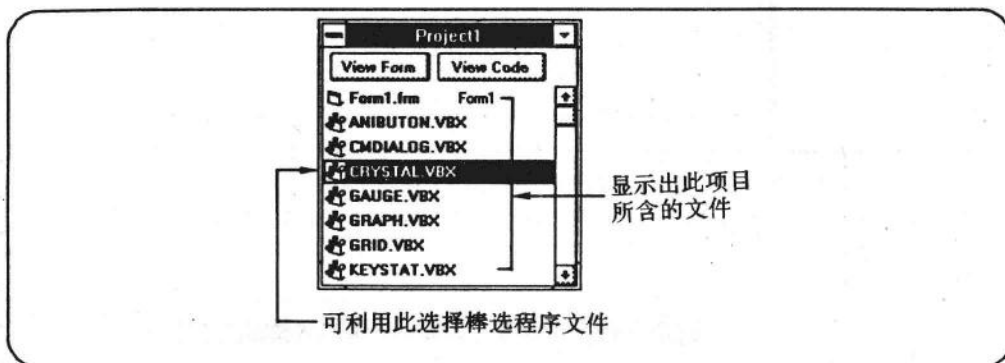
 : 显示当前调用(active call)的结构,你也可以执行 Debug 菜单的 Calls 命令,执行此钮的功能。

 : 在编码窗口(Code Window)内,一次执行一条命令。你也可以执行 Debug 菜单的 Single Step 命令,执行此钮的功能。

 : 在编码窗口(Code Window)内,一次执行一个过程(procedure)或一条命令。你也可以执行 Debug 菜单的 Procedure Step 命令,执行此钮的功能。

2. 项目窗口(Project Windows)

以前在 DOS 环境中,通常是一个程序文件即可完成所要的功能。不过在 Windows 环境中,若想设计一个功能,可借用许多程序文件完成,为了方便管理这些程序文件,可借用一个项目文件(Project file)管理它们。而项目窗口则是列出当前项目文件包含哪些程序文件。

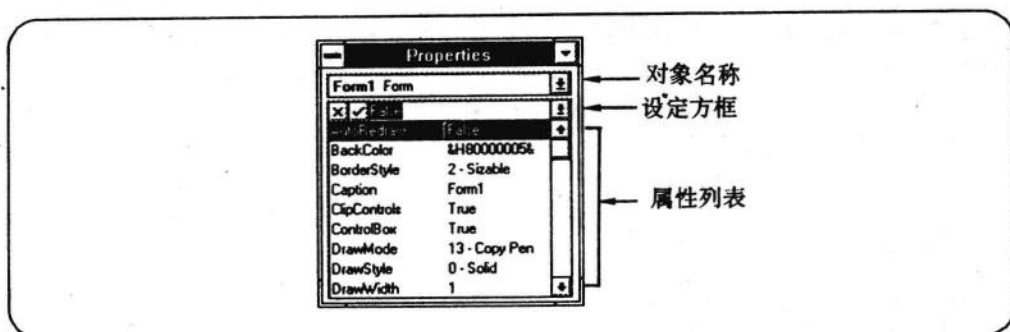


上述项目窗口内包含一系列的文件,同时有一个菜单,你可以利用选择棒选择某个文件,然后执行下列两个功能钮。

①View Form 钮:若是以实线显示,则表示按此钮可促使在屏幕上显示选择棒所选窗体文件的窗体(form)。若是此钮以虚线显示,则表示当前选择棒所选的并不是窗体文件(注:以 frm 为扩展文件名的文件称窗体文件),所以无法启用此钮。

②View Code 钮:若以实线显示,则表示按此钮可在屏幕上显示所需文件的程序码。

3. 属性窗口(Properties Window)



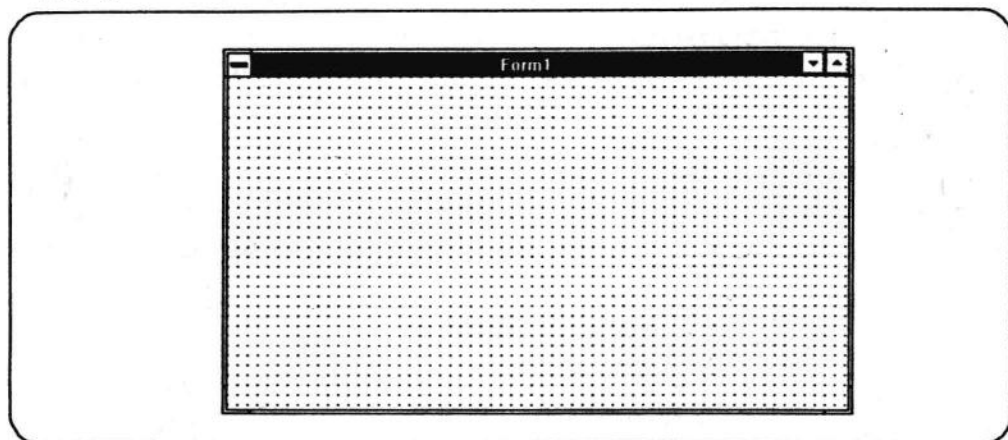
属性窗口包含三个部分：

①名称栏位：此栏位列出窗体(form)或是控件(control, control 的意义将在窗体(form)窗口说明)的名称。

②设定方框：列出属性列表内当前所选的属性。

③属性列表：列出当前所选的窗体或控件包含的所有的属性。

4. 窗体窗口(Form)



上述的窗体窗口(Form)，简称是窗体，而标题栏的 Form1 是此窗体的名称。在以 Visual Basic 设计 Windows 应用程序时，通常都是先在上述窗体内利用工具箱(下一主题将说明)设计用户介面，如下所示：

