

流行软件速成丛书

Visual Basic 使用速成

沈纪新 江 英 编



清华大学出版社

(京)新登字 158 号

内 容 简 介

Visual Basic 是开发 Windows 应用程序的一种简单易学的工具,它受到广大计算机用户的欢迎,得到了广泛的应用。

本书以最新版 Visual Basic 3.0 为背景,全面介绍了 Visual Basic 的基本概念和基本用法,是为初学者编写的速成书。它可作为各种培训班、广大计算机用户的教材或参考书。

JS/22/28

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

Visual Basic 使用速成/沈纪新,江英编著. —北京:清华大学出版社,1995

(流行软件使用速成)

ISBN 7-302-01983-5

I. V... I. ①沈... ②江... III. BASIC 语言-基本知识 IV. TP312BA

中国版本图书馆 CIP 数据核字(95)第 17040 号

出版者:清华大学出版社(北京清华大学校内,邮编 100084)

印刷者:北京密云胶印厂印刷

发行者:新华书店总店北京科技发行所

开 本:787×1092 1/16 印张:15.75 字数:367 千字

版 次:1995 年 11 月第 1 版 1995 年 11 月第 1 次印刷

书 号:ISBN 7-302-01983-5/TP·914

印 数:00001—10000

定 价:16.50 元

8800180

《流行软件速成丛书》编辑委员会

主 编	沈金发	
副主编	郑甫京	熊桂喜
委 员	王 敏	王国印
	李幼哲	郑全战

《流行软件速成丛书》序

流行软件涉及面最广,使用人数最多,体现了软件研究的最新成果,反映了计算机应用的热点所在。对于每个现代人,要想成功,要想在激烈的竞争中立于不败,熟练地使用相关的流行软件已是必然趋势。

对于接触计算机不多的人们来说,让他们一下子去读大厚本的手册或教材,就像进入一个全然陌生的世界,障碍重重,只会事倍功半。即使是具有计算机基础知识的人们,当他们面对日新月异的流行软件时,也会惶然不知所措。大量的抽象概念,严格的规范和操作规定;全新的用户界面,日益庞大的功能;编程思想的改变,开发机制的更新……所有这一切都增加了学习的困难。这就在计算机技术的迅猛发展与计算机技术迅速普及之间形成了一道令人烦恼的鸿沟。

《流行软件速成丛书》正是为解决这个难题而推出的。它是为我国千千万万的计算机用户架起一座方便快捷的桥梁,带着读者轻松而快速地走进计算机应用的最新领域。

《流行软件速成丛书》是独树一帜的,它几乎包括了当前流行软件的所有热门技术和热门课题。从使用到编程,从语言到工具,都编入丛书。我们将紧随软件的不断发展,配合最新、最流行、最实用的软件,不断推出新书目奉献给广大的读者。

《流行软件速成丛书》力求以通俗流畅的语言,在短而精的篇幅中把最实用、最重要的知识讲清楚、讲透彻,引导读者轻松入门,迅速应用,在应用中精通。

愿《流行软件速成丛书》能成为广大读者的知音和朋友,并衷心地希望广大读者对本丛书的不足或缺点提出批评,对今后的发展提出宝贵意见。

丛书编委会

1995年6月

前 言

许多人认为,Windows 预示着业余程序员的末日。在 MS-DOS 环境中,非计算机专业的人员一般都可以学会足够的编程知识,编写简单的应用程序,以解决工作中的问题。但在 Windows 中,最简单程序的编程工作都这么复杂,非计算机领域的专业人员还能进行编程工作吗?

1991 年,Microsoft 公司推出 Visual Basic,为那些心灰意冷的程序设计人员重新点燃了希望之火,使开发 Windows 应用软件不再让人望而却步。使用 Visual Basic 开发软件不但不需要 C 语言背景,甚至没有任何经验的初学者亦可愉快胜任,因为它的确是易学好用的。

难怪在 1991 年 10 月出版的 BYTE 杂志中,O. Linderholm 对 Visual Basic 作了专门介绍,认为它是取代 C 或 C++ 的最佳语言之一。Visual Basic 在 1991 年 BYTE 杂志的软件评优中荣获大奖。

因此,对于我国广大的 Windows 开发者,无论是专业程序设计人员还是一般的程序设计人员,迅速掌握 Visual Basic 已成为刻不容缓的需要。

我国已出版了少量的 Visual Basic 图书,但大多是翻译的外文资料,这显然不能满足中国读者的需要。而且在已出版的 Visual Basic 图书中,很少有结合我国的实际为初学者而写的,艰涩的术语、缺少中文注释的代码,使初学者在学习中发生困难。

本书正是为饥渴的初学者而作,以轻松流畅通俗自然的语言,将您领入 Visual Basic 的世界,尽览 Visual Basic 的迷人风采!

本书以最新版的 Visual Basic 3.0 为背景,全面介绍 Visual Basic 的基本概念和基本用法。由于是速成读物,所以本书不打算覆盖 Visual Basic 的每个角落(这也是没有必要的),而是集中阐述 Visual Basic 最常用、最实用的部分,讲清楚,讲透彻,决不是蜻蜓点水、雾里看花。

本书不是一般的入门读物,相反,它系统地、全面地介绍 Visual Basic 的基本知识,有些地方甚至还很深入。例如,对象坐标系是 Visual Basic 最重要的基本概念之一,但由于这个问题不易说清,所以大多数 Visual Basic 图书对此要么避而不谈,要么含含糊糊地躲闪而过。但是,本书从实用角度出发,透澈地讲解了对象坐标系,为读者的进一步学习扫除绊脚石。所以,本书对各种层次的读者都是有所裨益的。

总之,本书的出版必定为读者的实际开发工作和进一步的学习奠定了坚实的基础!

目 录

第 1 章 您好! Visual Basic	1
1.1 安装和启动 Visual Basic	1
1.2 HELLO 程序	2
1.2.1 保存新项目	3
1.2.2 项目窗口	4
1.2.3 属性窗口	4
1.3 附加 HELLO 程序的代码	6
1.4 运行 HELLO 程序	8
1.5 事件过程	8
1.6 改进 HELLO 程序	9
1.6.1 工具箱窗口	9
1.6.2 附加代码	13
1.6.3 运行改进版的 HELLO 程序	13
1.7 建立可执行文件	14
第 2 章 程序设计基础	15
2.1 数据类型	15
2.1.1 标准数据类型	15
2.1.2 用户自定义数据类型	16
2.2 常量和变量	17
2.2.1 命名规则	17
2.2.2 说明变量	18
2.3 运算符和表达式	19
2.3.1 算术运算符	19
2.3.2 关系运算符	20
2.3.3 逻辑运算符	21
2.4 基本语句	22
2.4.1 赋值语句	22
2.4.2 IF 语句	23
2.4.3 Select Case 语句	26
2.4.4 For 循环	28
2.4.5 Do 循环	29
第 3 章 窗体和控制	32

3.1	窗体	32
3.1.1	窗体的基本组成	32
3.1.2	改变窗体的大小	33
3.1.3	窗体的 BorderStyle 属性	35
3.2	单选钮、选择框和框架	36
3.2.1	缺省按钮和取消按钮	36
3.2.2	单选钮和选择钮	40
3.2.3	尺寸柄	43
3.2.4	字体和字形	44
3.2.5	框架	52
3.3	文本框和标签	56
3.3.1	标签用作输出	62
3.3.2	检查输入数据的合法性	65
3.3.3	多行文本	66
3.4	水平滚动条和垂直滚动条	68
3.5	通用过程	72
3.6	列表框和组合框	78
3.6.1	建立列表框	79
3.6.2	删除表项	82
3.6.3	清除表项	87
3.6.4	组合框	88
3.7	图片框和图象控制	89
3.7.1	在运行时装入图片	91
3.7.2	图象控制	92
3.7.3	Static 语句	97
3.7.4	Move 方法	99
3.7.5	图片框	107
3.8	计时器	108
3.9	动画	115
3.10	网格和 OLE 客户控制	121
3.10.1	网格控制	122
3.10.2	OLE 客户控制	126
第 4 章	菜单	128
4.1	基本概念	128
4.2	菜单设计窗口	129
4.3	附加代码	134
4.4	分隔条	136
4.5	快捷键	137

4.6	菜单控制的可见性	138
4.7	检取标记	138
4.8	编辑器	139
第5章	绘图.....	144
5.1	坐标系	144
5.2	画点	146
5.3	画线	153
5.4	画矩形	161
5.5	画圆	167
5.6	画圆弧和扇形	174
5.7	画椭圆	182
5.8	画椭圆弧	187
5.9	AutoRedraw 属性	187
5.10	小结.....	188
第6章	鼠标和键盘.....	190
6.1	鼠标事件	190
6.2	位置参数	191
6.3	按钮参数	194
6.4	换档参数	207
6.5	拖放	208
6.6	键盘事件	211
第7章	其它话题.....	213
7.1	文件系统	213
7.1.1	驱动器列表框.....	213
7.1.2	目录列表框.....	214
7.1.3	文件列表框.....	218
7.2	多重窗体	219
7.2.1	启动窗体.....	224
7.2.2	模块.....	224
7.3	数组	225
附录A	请求帮助.....	228
附录B	属性、事件、方法	231

第 1 章 您好! Visual Basic

从现在开始,我们将学习用 Visual Basic 编写 Windows 应用程序。本章将构造第一个 Visual Basic 应用程序 HELLO。通过 HELLO 程序的构造过程,读者将逐步熟悉 Visual Basic 的基本界面和基本术语,并对 Visual Basic 应用程序的构造过程有一个总体的认识。

1.1 安装和启动 Visual Basic

和大多数 Windows 应用程序一样,Visual Basic 有一个名叫 SETUP.EXE 的安装程序,它负责将 Visual Basic 安装到硬盘上。

按下述步骤安装 Visual Basic:

- 在 MS-DOS 提示符下键入:

WIN✓

以启动 Windows。

- 把 1 号磁盘 Disk 1 插入 A 驱动器(或 B 驱动器)。
- 启动 Windows 的 File Manager(文件管理器)从 File 菜单选择 Run 命令。Windows 于是显示 Run 对话框,如图 1.1 所示。
- 如果 Disk 1 插在 A 驱动器中,则在 Run 对话框中键入:
A:\SETUP✓
如果 Disk 1 插在 B 驱动器中,则在 Run 对话框中键入:
B:\SETUP✓
- 遵循 SETUP 程序给出的指示,完成安装过程。

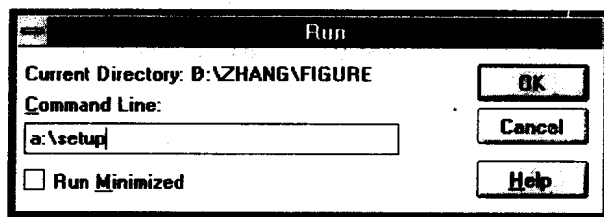


图 1.1 Run 对话框

可用下述两种方法启动 Visual Basic:

1. 在 Windows 的 Program Manager(程序管理器)中双击 Visual Basic 3.0 图标(将鼠标指针放在该图标之上,再快速连接两下鼠标左按钮)。Windows 于是显示 Visual Basic 3.0 应用程序组,如图 1.2 所示。双击 Microsoft Visual Basic 图标。

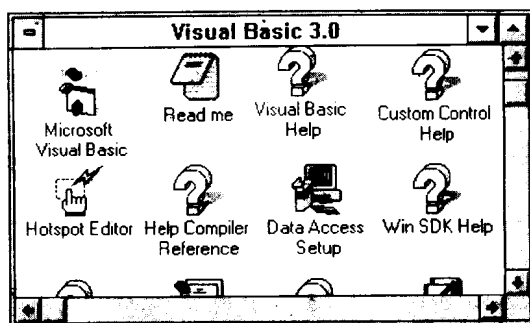


图 1.2 Visual Basic 3.0 应用程序组

2. 在 MS-DOS 提示符下,键入:

WIN VB\VB

1.2 HELLO 程序

一个 Visual Basic 应用程序叫做一个项目(project),它由多个文件组成。下面为 HELLO 程序建立一个新项目:

- 启动 Visual Basic。(Visual Basic 的显示如图 1.3 所示。)
- 单击 File 菜单,把鼠标指针移到 File 菜单标题上,按一下鼠标左按钮。Visual Basic 于是打开 File 菜单。
- 单击 New Project 菜单项。

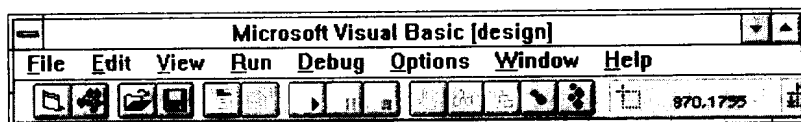


图 1.3 启动后的 Visual Basic

Visual Basic 于是显示各种窗口,其中有一个窗口的标题是 Form1,如图 1.4 所示。这就是在 Visual Basic 中作为用户界面的程序窗口:窗体(Form)。图 1.4 所示的窗体是个空白窗口,我们所要做的是在窗体中加入各种 Visual Basic 工具。

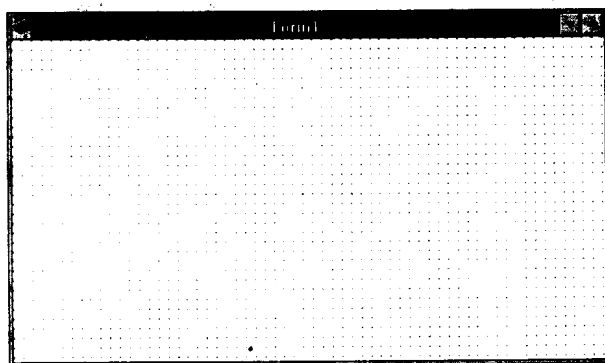


图 1.4 空白窗体

1.2.1 保存新项目

我们迄今还没有做任何实质性的工作,但不妨先保存这个新项目。

前面说过,一个项目由多个文件组成。具体说来它包括以下两种文件:

1. make 文件。make 文件的扩展名为 .MAK。一个项目只有一个 make 文件。Visual Basic 根据 make 文件提供的信息构造项目。

2. 窗体文件。窗体文件的扩展名为 .FRM。一个项目可有多个窗体文件。HELLO 程序是个单窗体项目,所以只包含一个窗体。

按下述步骤保存 HELLO 项目:

- 从 File 菜单选择 Save Project As。Visual Basic 于是显示如图 1.5 所示的对话框,询问是否保存窗体文件 Form1.frm。

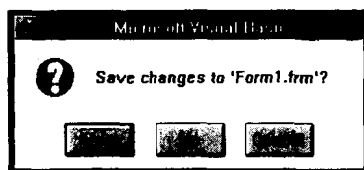


图 1.5 对话框

- 单击 Yes 按钮,表示想保存窗体。Visual Basic 于是显示 Save File As 对话框,如图 1.6 所示。Save File As 对话框用来设置文件名和文件所在的目录。读者可通过在 Directories 列表框中双击目录来选择合适的目录。File Name 框中是缺省的文件名,由于缺省名 Form1.frm 很不直观,所以应重新命名。

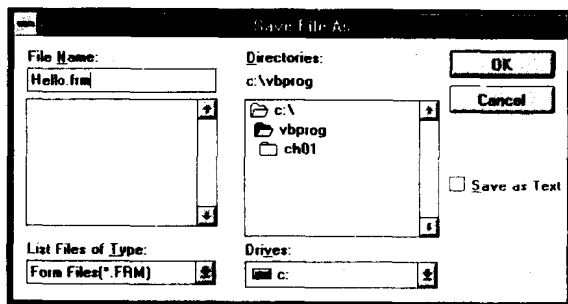


图 1.6 Save File As 对话框

- 在 File Name 框中键入 HELLO.FRМ,然后单击 OK 按钮。在图 1.6 中把 HELLO 项目的窗体文件以 HELLO.FRМ 名保存到 C:\vbprog\CH01 目录中了。Visual Basic 接着显示 Save Project As 对话框。Save Project As 对话框用来指定 make 文件的文件名及其所在的目录,如图 1.7 所示。
- 选择合适的目录(应和窗体文件 HELLO.FRМ 所在的目录一致)。
- 在 File Name 框中键入 HELLO.MAK,以取代原来的缺省名 Form1.mak。

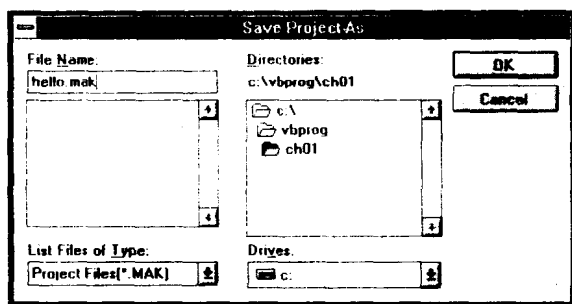


图 1.7 Save Project As 对话框

1.2.2 项目窗口

这样就构造了一个名叫 HELLO.MAK 的项目,它只包含单个窗体文件。

■ 从 Windows 菜单选择 Project。Visual Basic 于是显示如图 1.8 的项目窗口。

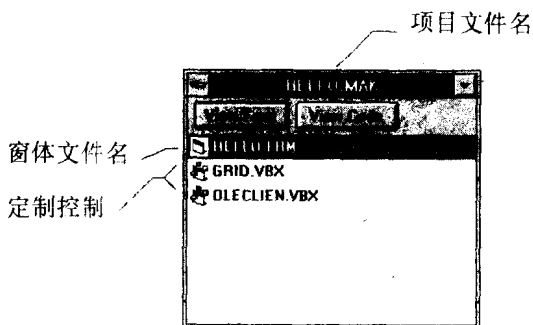


图 1.8 项目窗口

项目窗口的标题是项目文件名 HELLO.MAK,窗口中列出了一系列文件,其中 HELLO.FRM 是项目的窗体文件。项目窗口中有两个按钮:View Form 按钮和 View Code 按钮,分别用于查看窗体和查看代码窗口。

1.2.3 属性窗口

在项目窗口中,窗体文件名 HELLO.FRM 的右边显示着 Form1,Form1 是该窗体的名字。窗体是一个程序窗口,它有自己的名字、长、宽、背景色等性质,这些性质叫做窗体的属性(properties)。可以通过属性窗口设置属性。

■ 从 Windows 菜单选择 Properties。Visual Basic 于是显示属性窗口,如图 1.9 所示。

属性窗口由对象框、设置框和属性列表组成。属性列表的左列是一系列属性,如 AutoRedraw、BackColor 等,右列是各属性的缺省设置,也就是 Visual Basic 自动设置好的值。

对象框用来指示当前显示的是哪个对象的属性列表。“对象”是个面向对象的术语,例如窗体、命令钮、滚动条等都是对象,Visual Basic 程序处理的正是一个个对象。属性用来

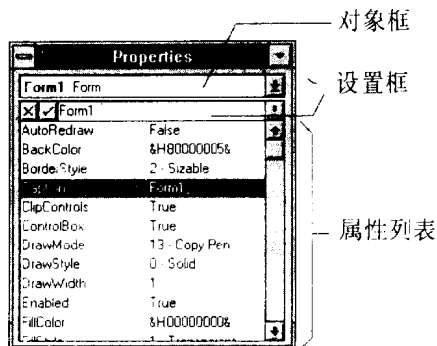


图 1.9 属性窗口

定义对象的外观和行为。例如窗体是一个对象,属性 Caption 指定窗体的标题,该标题显示在窗体窗口的标题标栏上。BackColor 也是窗体的属性它指定窗体的背景色。

在图 1.9 所示的属性窗口中,对象框中的文本内容是“Form1 Form”,表明这是一个名叫 Form1 的窗体的属性窗口。Form1 正是图 1.4 所示的窗体,该窗体的标题 Form1 显示在窗体窗口上的标题栏上。请注意,Name(名字)属性和 Caption(标题)属性是两个不同的属性,对象框中显示的是名字,属性列表中 Caption 属性右边显示的是标题,Name 属性右边显示的是名字。

下面改变 Form1 窗体的 Caption 属性:

- 单击属性列表中的 Caption 属性。Visual Basic 于是加亮 Caption 属性,并在设置框中显示 Caption 属性的当前值 Form1。设置框用来设置属性的值。
- 在设置框中键入 The Hello Program,按回车键。如图 1.10 所示。Visual Basic 于是在 Form1 窗体的标题栏上显示 The Hello Program。

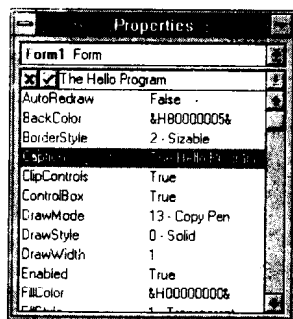


图 1.10 改变 Form1 窗体的 Caption 属性

下面设置 Form1 窗体的 BackColor 属性:

- 单击属性列表中的 BackColor 属性。
- 单击设置框右端的三个圆点,如图 1.11 所示。Visual Basic 于是显示调色板窗口,如图 1.12 所示。
- 在中意的颜色块上单击一下。Visual Basic 于是相应地改变 Form1 窗体的背景色。

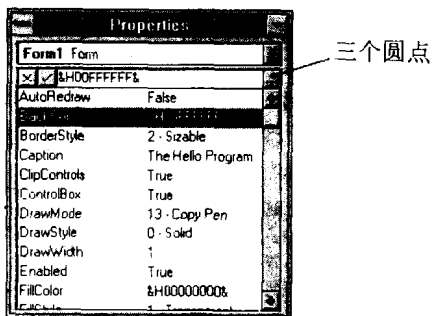


图 1.11 选择 BackColor 属性

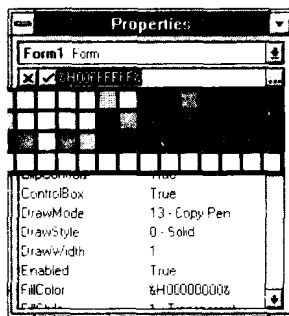


图 1.12 调色板窗口

设置对象的属性一般有三种方式：

1. 设置框右端的向下箭头为暗淡色时，表明该箭头无效，这时只能直接键入所需的值。例如设置 Caption 属性时，必须直接键入想要的标题文本。

2. 设置框右端显示三个圆点时，一般单击这三个圆点。Visual Basic 将打开一个窗口或对话框，用户可在其中选择所需的设置。

3. 当设置框右端的向下箭头正常显示时，可以通过单击该箭头来打开一个下拉式列表，用户在列表中选择所需的值。如图 1.13 所示。这种情况下也可以直接在设置框中键入所需的值，但键入的值只能是下拉式列表中出现的设置。

前面曾提到 Form1 窗体的 Name 属性。下面改变 Name 属性：

■ 在属性窗口中单击属性列表右边的垂直滚动条的向下滚动箭头，直到显示出 Name 属性，如图 1.13 所示。Visual Basic 于是向上滚动属性列表。

■ 单击属性列表中的 Name 属性。

■ 在设置框中键入 frmHello。

这样就把该窗体改名为 frmHello。

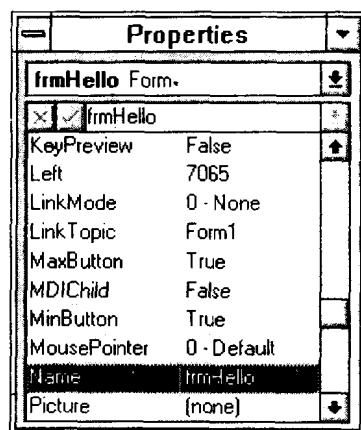


图 1.13 Name 属性的设置

1.3 附加 HELLO 程序的代码

在前面我们认识了 Visual Basic 的程序窗口、窗体，并通过属性窗口设置了窗体的某些属性，这时窗体的外观和最初显示的空白窗体已有所不同了。等会儿运行 HELLO 程序时，读者将发现，运行时的窗体和此时处于设置状态的窗体外观上是一样的。在设计阶段就可以预见到运行时的窗体外观。这就是“可视”(visual)的含义。

Visual Basic 应用程序的构造过程一般分为两个阶段：

1. 可视实现阶段；

2. 附加代码阶段。

在可视实现阶段不需要编写任何代码,所要做的只是在窗体中加入各种 Visual Basic 工具,例如命令钮、文本框等控制。并且通过属性窗口设置这些对象的属性。我们前面所做的工作就是 HELLO 程序的可视实现,只是 HELLO 程序的可视实现非常简单,没有在窗体中加入各种控制。

可视实现阶段结束时,已经可以预见到运行时的窗体外观,“可视实现”由此得名。

在附加代码阶段,编写并键入程序代码。下面为 HELLO 程序附加上的代码。

- 单击项目窗口的 View Code 按钮。或者在 FrmHello 窗体内双击。Visual Basic 于是显示 HELLO 程序的代码窗口,如图 1.14 所示。有些情况下,项目窗口的 View Code 按钮为暗淡色,这时应先单击项目窗口中的窗体文件名 HELLO.FRM,然后再单击 View Code 按钮。

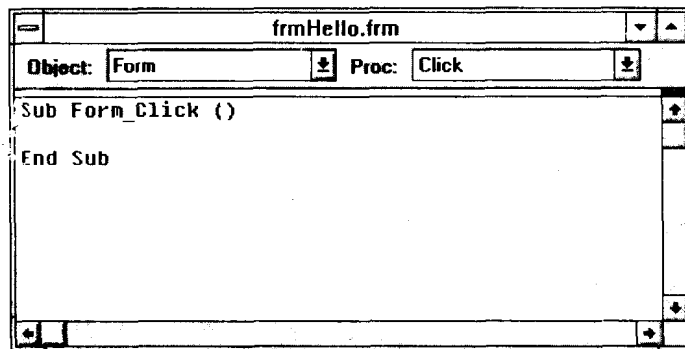


图 1.14 代码窗口

- Object(对象)框右端的向下箭头。Visual Basic 于是下拉对象列表框。
- 在打开的下拉式列表框中单击 Form 项。Object 框中于是显示 Form。
- 单击 Proc(过程)框右端的向下箭头。Visual Basic 于是下拉事件列表框。
- 在打开的下拉式列表框中单击 Click 项。Proc 框中于是显示 Click。代码窗口中于是显示事件过程:

```
Sub Form_Click()
```

```
End
```

- 在事件过程 Form_Click()中键入下述代码:

```
Sub Form_Click()  
    Print "Hello, Visual Basic"  
End Sub
```

- 从 File 菜单选择 Save Project,保存所做的工作。

1.4 运行 HELLO 程序

- 从 Run 菜单中选择 Start。Visual Basic 于是运行 HELLO 程序,并显示窗体窗口。
- 在窗体内单击一下。Visual Basic 于是显示“Hello, Visual Basic”,如图 1.15 所示。

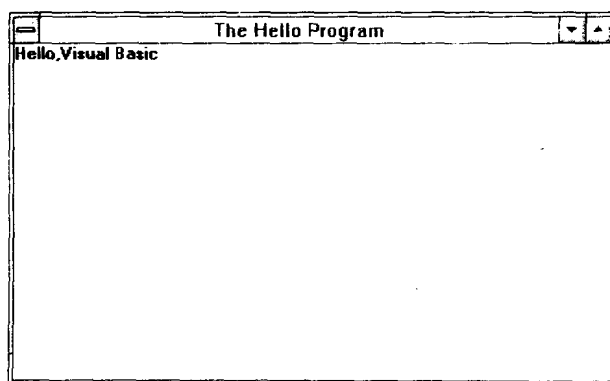


图 1.15 运行时的 HELLO 程序

- 不断单击窗体。Visual Basic 依次相应显示“Hello, Visual Basic”。
- 从 Run 菜单中选择 End,结束 HELLO 程序。

1.5 事件过程

下面分析 HELLO 程序的工作原理。事件过程 From_Click()如下所示:

```
Sub Form_Click()  
    Print "Hello, Visual Basic"  
End Sub
```

其中最第一行和最后一行是 Visual Basic 已经编写好了的,读者所键入的只是中间一行。

关键字 Sub 表示着一个过程的开始。From_Click()是事件过程名,右边()是空的参数表。其中 Form 和 Click 分别是指“窗体”和“单击”,意思是每当用户单击窗体时,事件过程 From_Click()就会自动执行。

Visual Basic 应用程序首先是 Windows 应用程序,是由 Windows 操作系统管理并支持的。可以把 HELLO 程序的执行过程用形象化语言描绘如下:刚开始运行时,HELLO 程序就躺着睡大觉,Windows 却精神抖擞地四处查看动静。当用户单击 frmHello 窗体时,Windows 立刻踢醒 HELLO 程序,通知它:“用户单击 frmHello 窗体了!”。HELLO 程序当即清醒过来,并琢磨“frmHello 窗体(From)被单击(Click)……我应该执行事件过程 From_Click()!”,HELLO 程序于是执行 From_Click(),执行完后又埋头大睡,直到 Windows 再次踢醒它……

用专业用语描述上述过程:开始运行时 HELLO 程序被挂起,当用户单击 frmHello

窗体时,“单击”事件发生,Windows 于是把相应的信息传给 HELLO 程序,触发事件过程 From_Click()的执行。

需要注意的是:在从前的程序设计语言中,函数和过程总是被调用执行,而在 Visual Basic 应用程序中,总是事件触发事件过程的执行。换句话说,事件过程响应已发生的事件而自动执行,这里“自动”是和“被调用”意思相对的。Windows 操作系统在整个过程中做了很多工作。

那么事件过程 From_Click()到底做了什么呢?

From_Click()过程执行语句

```
Print "Hello,Visual Basic"
```

这是一个 Print 语句,在窗体上打印文本 Hello,Visual Basic 双引号标识文本串的开始和结束。

最后一行的 End Sub 表示看该过程的结束,它和第一行的 Sub 相匹配。

还有一个问题需要说明:当程序中有多个窗体时,From_Click()到底是哪个窗体的事件过程呢?如图 1.14 所示,代码窗口的标题栏上已清楚地表明这是窗体文件 HELLO.FRM 的代码,所以该 From_Click()事件过程对应 frmHello 窗体。

1.6 改进 HELLO 程序

HELLO 程序的用户界面仍然是个空窗体,下面改进 HELLO 程序,在窗体中加入各种控制。

1.6.1 工具箱窗口

- 从 Windows 菜单选择 Project。Visual Basic 于是显示项目窗口。
- 单击项目窗口的 View Form 按钮。Visual Basic 于是显示窗体。
- 从 Windows 菜单选择 Toolbox。Visual Basic 于是显示工具箱窗口,如图 1.16 所示。

工具箱窗口中包含各种控制的图标,我们所要做的是选择合适的控制,并把它们放入窗体中。

图标(icon)是一种小的图形图象,可用来代表最小化的程序窗口、一个控制甚至一个命令。控制(control)是 Visual Basic 已经预定义的子窗口,它能完成指定类型的功能。标准版的 Visual Basic 3.0 在工具箱窗口中显示了 20 种控制的图标,如图 1.16 所示。专业版的 Visual Basic 3.0 提供了更多的定制控制,如下所述:

1. 指针(Pointer) 这是工具箱窗口中唯一不能用来绘制控制的工具,只起到指示作用。
2. 图片框(Picture Box) 用于显示图形图象、接收图形方法的输出,还可用来容纳其它控制。
3. 标签(Label) 用于显示和输出文本信息,但不能用作输入界面。
4. 文本框(Text Box) 用于输入和输出文本。