



以例说文 强调实际操作
通俗易懂 注重现学现用

例说

张敬华 编著

Visual Basic 6.0

北京大学出版社
<http://cbs.pku.edu.cn>

内 容 简 介

Visual Basic 6.0是Microsoft公司推出的功能强大的Windows编程语言。本书通过大量的实例,由浅入深、循序渐进地介绍了Visual Basic 6.0的使用方法。首先是Visual Basic 6.0的基础编程,可以将这部分作为初级读者的必读章节。然后本书按程序分类将后面的Visual Basic 6.0编程实例划分为窗体设计、图形处理、单/多文档编程、多媒体程序设计、数据库编程、最热门的网络编程、鼠标事件处理及响应和应用程序发布等。

通过本书的学习,既可以使 Visual Basic 6.0 的新用户进入 Visual Basic 6.0 的大门,又可以使 Visual Basic 6.0 的老用户得到进一步提高。。

图书在版编目(CIP)数据

例说 Visual Basic 6.0/张敬华编著. —北京:北京大学出版社, 2000.4
(流行软件例说丛书)
ISBN 7-301-01835-5

I. 例… II. 张… III. BASIC 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2000) 第 07028 号

书 名: 例说 Visual Basic 6.0

著作责任者: 张敬华

责任编辑: 黄庆生 汉 明

标准书号: ISBN 7-301-01835-5/TP·141

出 版 者: 北京大学出版社

地 址: 北京市海淀区中关村北京大学校内 100871

网 址: <http://cbs.pku.edu.cn>

电子信箱: xxjs@pup.pku.edu.cn

排 版 者: 南方立德 (Leader) 信息技术中心

印 刷 者: 河北省滦县印刷厂

发 行 者: 北京大学出版社

经 销 者: 新华书店

787 毫米×1092 毫米 16 开本 20.625 印张 502 千字

2000 年 6 月第 1 版 2000 年 6 月第 1 次印刷

定 价: 32.00 元

前 言

比尔·盖茨曾对他的编程人员说：“你们有不能解决的问题都交给我，让我用 Visual Basic 来实现！”

“没有 Visual Basic 不能实现的程序效果”，我想这大概是对 Visual Basic 的最好注释！在很长的一段时间，程序员不得不为了实现程序目的而编制大量难懂的代码，如众所周知的 C/C++ 系列。当然，笔者对 C/C++ 的高效并没有偏见！但是，如果你想要精通这种很“高深”的工具并不是一件很容易的事，你可以在编制 C 程序时明显的感觉到我们所说的“高深”。如果说这就是聪明的程序员选择 Visual Basic 的原因，我想谁都愿意相信！但 Visual Basic 提供给我们的远不止这些。

打开 Visual Basic 的编辑界面，试着编制一个程序。你可以发现一切高深的技术细节都封装在了控件的内部，所有你想实现的程序都可以使用这种高度封装的办法来实现。同样，在 Visual Basic 中，你可以使用和 C/C++ 一样的办法实现前面所说的高效——调用系统 API 函数，你所需要的几乎都可以在这里找到，因为，我们正在使用的 Windows 就是这样工作的。也就是说，在 Visual Basic 中，你可以和系统同步地调用所有的动态链接库，完全公平的调用！

在 Visual Basic 中，用不着像在其他的编程工具中那样为了调整程序的结构让它更清晰而伤神。因为 Visual Basic 所提供的程序模块化就已经解决了这种也许是一直困扰程序员的问题。所有的程序都将按照不同的对象来划分结构，如果你能准确地给你所使用的控件起一个你所能记住的名称，我想在程序结构上就应该不会有你曾经有过的那种问题——结构混乱，Visual Basic 已经为你解决了这个问题！

Visual Basic 6.0 是 Microsoft 公司推出的 Visual Basic 的最新版本。本书通过大量的实例，由浅入深、循序渐进地介绍了 Visual Basic 6.0 的使用方法。首先是 Visual Basic 6.0 的基础编程，读者可以将这部分理解为初级读者的必读章节。然后本书使用按程序目分类的方式将后面的 Visual Basic 6.0 编程实例划分为窗体设计、图形处理、单/多文档编程、多媒体程序设计、数据库编程、最热门的网络编程、鼠标事件处理及响应和应用程序发布等。

本书由孙景利策划，张敬华主编，另外，陆谊、丁雨、薛建林、王乐、马明、汪兵、黄少棠、瞿小玉、黄瀚华、凌贤伍、胡梦霞、姚玉霞、孙敬娜、付红梅、康孟霞、张小东、李宁、王强、赵四海、李晓峰、董团结、杨仕润、韩百、涂海滨等也参加了全书的编写工作，在这里对他们表示诚挚的感谢。

由于时间紧迫，加之笔者水平有限，书中如有疏漏之处，请广大读者批评指正。

编 者

2000 年 5 月

目 录

第 1 章 VB 6.0 的开发环境.....	1
1.1 关于 Visual Basic	1
1.2 VB 安装中的小问题.....	1
1.3 VB 的集成开发环境.....	2
1.4 一章回顾	10
第 2 章 Visual Basic 的语法	11
2.1 Visual Basic 应用程序的结构	11
2.2 变量、常数和数据类型概述	14
2.3 数组	23
2.4 过程	24
2.5 关于对象	30
2.6 一章回顾	31
第 3 章 基本控件介绍	32
3.1 关于控件的小常识	32
3.2 常用的基本控件	33
3.3 一章回顾	55
第 4 章 窗体设计	56
4.1 启动窗体	56
4.2 窗体的属性、事件、方法	58
4.3 窗体中的菜单设计	59
4.4 窗体中的工具条、状态条的设计	67
4.5 一章回顾	85
第 5 章 图形处理	86
5.1 使用图形控件	86
5.2 绘图方法	95
5.3 图形处理	101
5.4 一章回顾	107
第 6 章 SDI 和 MDI 应用程序的设计	108

6.1 创建一个 SDI 应用程序	109
6.2 创建一个 MDI 应用程序	137
6.3 一章回顾	159
第 7 章 多媒体	160
7.1 VB 实现多媒体的方法	160
7.2 现在开始	165
7.3 一个综合的多媒体编程示例	179
7.4 一章回顾	193
第 8 章 数据库	194
8.1 VB 对数据库的支持	194
8.2 数据库基础	195
8.3 VB 的数据库控件	205
8.4 使用 VB 的数据库控件	212
8.5 使用数据访问对象和数据库	224
8.6 一章回顾	252
第 9 章 网络编程	253
9.1 一点常识	253
9.2 常用控件	254
9.3 使用常用的控件编程	256
9.4 关于网络编程的综合实例	269
9.5 一章回顾	284
第 10 章 API 函数调用	286
10.1 API 函数的声明	286
10.2 使用 API 文本浏览器	290
10.3 几个有意思的 API 函数	293
10.4 API 实例介绍	298
10.5 再说 API	304
第 11 章 响应鼠标事件	305
11.1 简单的鼠标事件	305
11.2 拖放技术	311
11.3 也说鼠标	314
第 12 章 应用程序的发布	316
12.1 打包的方法	316
12.2 打包	317
12.3 展开我们的包	321

12.4 安装我们的应用程序	322
12.5 一章回顾	322

第1章 VB 6.0 的开发环境

1.1 关于 Visual Basic

在当今数字化的世界中，计算机的发展成为人们关注的焦点问题，计算机已深入到人们生活的各个领域。而使用计算机完成一定的工作，软件是必不可少的，软件又是通过编程来实现的，这样就促进了编程语言的飞速发展。从最开始的低级语言，如汇编语言，那时你必须了解许多有关硬件、系统的知识才能写出有用的应用程序。再到后来发展到一些高级语言，如 Pascal、C 等。这些高级语言为编程人员提供了许多极其方便的功能，如调用操作系统的一些功能等。这时你不用深入了解有关硬件、系统的专业知识，也能编出有用的程序。

现在大家普遍使用的第三代编程语言，如 Microsoft 公司的 Visual Studio 系列的开发工具 VC (Visual C++)、VB (Visual Basic)、VJ，Bland 公司的 Delphi、C++Builder 等都是面向对象的可视化的编程工具。这些编程工具的出现使编程人员受益匪浅，它们的共同特点是可视化的程度非常高，开发过程中，你面对的将不再完全是枯燥的程序代码，而且编写的应用程序中的大量繁杂的源代码也能够自动生成了。你只需为应用程序中的对象添加一些属性和编写一段事件代码。VB 是这些工具里最简单、最容易上手的，这样即使没有特别多的专业知识，从这本书学起，不出一个月你也可以编出许多实用好玩的东西。传统的应用程序开发有 3 个明显的步骤：编码、编译和测试代码。但是 VB 和它们不同，它使用交互式方法开发应用程序，使 3 个步骤间不再有明显的界限。

Microsoft 公司的 Visual Basic 开发大致经历了以下的发展过程：1991 年正式推出了 Visual Basic 1.0 版，经过了对 1.0 版本的修改后，Microsoft 紧接着推出了 Visual Basic 2.0 版本，在接下来的一年里，经过对 2.0 版本不断的修改和完善后又推出了 Visual Basic 3.0 版本。本着不断进取、完善的信念，他们在虚心、广泛地听取用户的意见后，于 1995 年又推出了 Visual Basic 4.0 版，1997 年发布了功能更强大的 Visual Basic 5.0 版。到 1998 年，现在被大多数程序员所认可的、非常优秀的 Visual Basic 6.0 版也正式诞生了。

1.2 VB 安装中的小问题

至于中文版的标准名称，Microsoft Visual Basic 共有 3 个版本：企业版、专业版和标准版。MSDN Library Visual Studio 6.0 版中所指的学习版就是 Microsoft Visual Basic 6.0 中

文标准版。

关于联机注册。在安装中文 VB 6.0 的过程中，安装向导会提示你通过 Web 注册。

关于安装数据访问应用程序的问题。当重新安装包含数据访问组件的 VB 6.0 应用程序时，如果 Windows 95 和 Windows 98 的 DCOM 没有出现在客户机的 Windows 9x 上，如果工程包含对 ADO、OLEDB 或 ODBC 的引用（可在向导的 Included 文件页中进行核实），可通过软件包和安装向导将文件 Mdac_typ.exe 增加到安装程序中。这一文件将 MDAC 2.0 文件安装到客户机上。

MDAC 2.0 正常工作需要 Windows 95 和 Windows 98 的 DCOM，然而在安装时它并不检测这一点。如果 Windows 95 和 Windows 98 的 DCOM 没有出现在客户机的 Windows 9x 上，安装会失败。一些旧的数据访问组件将在失败前被覆盖，这可能会引起客户机上原有数据访问应用程序执行失败。

1.3 VB 的集成开发环境

当你打开 VB 6.0 后，就会看到如图 1-1 所示的用户界面。

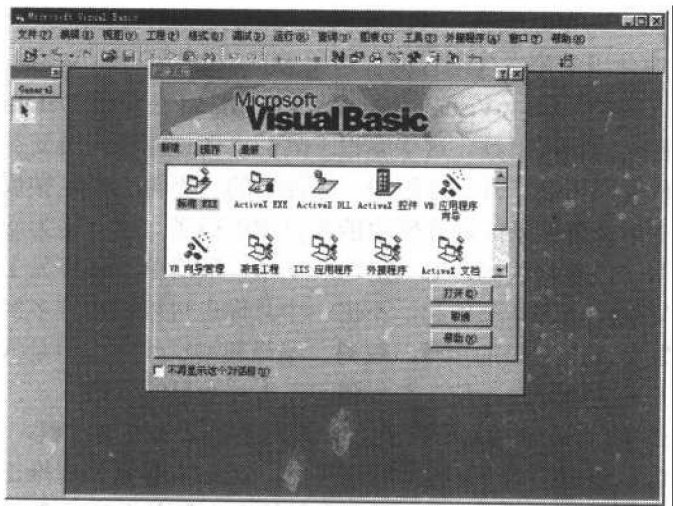


图 1-1 Visual Basic 的用户界面

它包括了为了快速有效地开发功能强大的 Windows 应用程序所需要的全部工具。通过它们的灵活应用可以开发出下列类型的应用程序，如表 1-1 所示。

表 1-1 Visual Basic 开发应用程序类型列表

类 型	说 明
Standard exe	标准 exe 程序——典型的 Windows 应用程序
Active exe	Activex 可执行应用程序
Activex dll	Activex 动态链接库

(续表)

类 型	说 明
Active 控件	Activex 控件
data project	数据工程
Iis application	Iis (internet information service) 应用程序
Addin ide	加入件
Activex document dll	Activex 文档动态链接库
Activex document exe	Activex 文档应用程序
Dhtml application	Dhtml (动态 html) 应用程序
vb enterprise editon controls	vb 专业版控件
vb application wizard	Vb 应用程序向导
vb wizard manager	Vb 向导应用程序

在 VB 的 IDE 集成环境中最上边是如图 1-2 所示的菜单条。

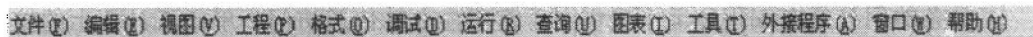


图 1-2 Visual Basic 的菜单条

菜单条囊括了绝大多数命令，你可以通过它们来控制编程环境。菜单条下边是工具条，上面的按钮都是菜单命令的快捷方式，你可以通过这些按钮更快地控制编程环境。屏幕上还将显示工具箱、工程容器窗口、窗体窗口、立即窗口、属性窗口以及窗体布局窗口，这些窗口的大小和样式依赖于系统配置。

现在我们来逐个简单地介绍一下这些窗体。先来介绍一下菜单，VB 中的菜单条除了提供了标准的“文件”、“窗口”、“编辑”、“视图”、“帮助”菜单项外，还提供了编程专用的功能菜单，如“工程”、“格式”、“运行”、“外接程序”、“图表”等，你会在今后的应用中渐渐地熟悉它们。

“文件”(File)：包含了一些对文件的基本操作命令，如新建、打开、移除、添加工程、保存、打印、生成工程 EXE 文件等。

“编辑”(Edit)：包含了与编辑有关的命令，如粘贴(Paste)、删除(Del)、查找(Find)、插入参数信息、属性方法列表等。

“视图”(View)：提供了一些显示和隐藏集成开发环境(IDE)元素的命令，如代码窗口(Code)、属性窗口(Properties Windows)、调色板、缩放等。

“工程”(Project)：提供了将 MDI 窗体、模块以及用户控件属性页、DHTML PAGE、DATA REPORT、WEB CLASS、文件等引入当前项目等命令。

“格式”(Format)：提供了对齐、顺序等命令。

“调试”(Debug)：提供了对程序进行调试的各个命令，如逐语句(Step into)、逐过程(step over)、添加监视(Add watch)等。

“运行”(Run)：提供了运行程序的命令，如启动(Start)、全编译执行等命令。

“查询”(Search)：提供了查询命令，如验证 SQL 语法、升降序排序、分组等。

“图表”(Chart)：提供了在项目中对图表操作的一些命令，如添加显示功能表、修改自定义视图等命令。

“工具” (Tools): 包含了启动菜单编辑器 (Menu Editor) 命令和对 IDE 环境进行配置 (Option) 的选项, 还包括了建立 ActiveX 构件和 ActiveX 控件所需的工具。

“外接程序” (Add-Ins): 缺省方式下提供了可视化数据管理器 (Visual data Manager) 外接程序 (Visdata), 编程人员可以通过外接程序管理器加入或删除外接程序。

“窗口” (Windows): 包含了设计窗口中各个窗体和代码窗口的布局命令, 如水平平铺 (Tile Horizontally)、垂直平铺 (Tile Vertically)、层叠 (Cascade)、排列图标 (arrange Icons) 等命令。

“帮助” (Help): 包含了 Visual Basic 的帮助信息。

菜单条的下边是工具条, 如图 1-3 所示。



图 1-3 Visual Basic 的工具条

工具条的用处是在编程环境下提供对于常用命令的快速访问。单击工具条上的按钮, 则执行该按钮所代表的操作。按照缺省规定, 启动 Visual Basic 之后显示“标准”工具条, 附加的编辑、窗体设计和调试的工具条可以用“视图”菜单上的“工具条”命令移进或移出。

在整个 IDE 开发环境的最左边是工具箱, 如图 1-4 所示。

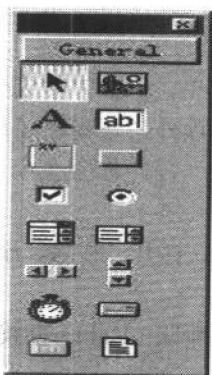


图 1-4 Visual Basic 的工具箱

它提供了一组用于设计窗体的放置控件。除了缺省的工具箱布局之外, 还可以通过在上下文菜单中选定“添加选项卡”, 并在结果选项卡中添加控件来创建自定义布局。下面让我们先来看看工具箱中的常用控件。

(1)  Command Button (命令按钮控件): 该控件在项目中的作用是在人机交互中提供确切的命令要求。命令要求通过编写代码来实现, 改变该控件的“Caption”属性将改变显示在命令按钮上的帮助说明文本。

(2)  Label (标记控件): 该控件显示用户不能编辑的文本。用该控件的“Caption”属性来设置要显示在窗体的文本, 在程序中通过更改“Caption”属性达到更改文本的目的。标记控件通常用来标识其他控件。

(3)  PictureBox (图形框控件): 该控件用来显示图形。用控件的“Picture”属性来设置所要显示的图形。

(4)  TextBox (文本编辑框): 该控件用于显示用户可以编辑的文本。与 Label 控件不同的是, 它实质是个微型文字编辑器, 利用该控件的“Text”属性, 可以设置控件上的文本或读取用户输入的文本。

(5)  Frame (帧控件): 该控件用于组合其他元素, 如复选框。

(6)  CheckBox (复选框控件): 复选框控件提供给用户若干个可以选择的选项。用户可以依次选择一个或若干个选项。程序通过判断控件的“Value”属性来判断用户是否选中该控件, 选中时, 控件的“Value”属性值为 1; 否则控件的“Value”值为 0。

(7)  Option Button (选项按钮控件): 选项按钮控件一般是成组出现的, 用户只能

选择其中一个选项，这一点与 CheckBox 控件不同。选定控件时，控件的“Checked”属性就为 True，否则，“Checked”属性为 False。

(8)  **ListBox** (列表控件): 该控件包含了一系列选项，用户可以选择其中的一个或几个。列表框控件中所选项目由控件的“Text”属性给定，而控件的“Sorted”属性确定了表中项目是否排序。

(9)  **ComboBox** (组合框控件): 组合框控件与 ListBox 控件类似，但不同的是组合框控件包含了一个文本编辑字段，用户既可以选择表中的项目，也可以在编辑字段中输入新字符串。利用该控件的“Text”属性就可以得到用户所选择的项目或输入的字符串。

(10)  **Horizontal ScrollBox** (水平滚动条控件): 用于输入一个数据，由在控件的最大值和最小值之间的位置确定用户输入值的大小。

(11)  **Vertical ScrollBox** (垂直滚动条控件): 其作用与水平滚动条一致，唯一的区别是滚动的形状不一样。

(12)  **Timer** (定时器控件): 用于完成确定时间间隔内必须进行的任务，用“Interval”属性确定事件发生的时间间隔。

(13)  **Drive ListBox** (盘列表框控件): 显示系统上的驱动器，用户在其中可以选择某个驱动器，如 C 盘、D 盘。

(14)  **Directory ListBox** (目录列表框控件): 显示当前驱动器上所有文件目录。

(15)  **File ListBox** (文件列表框控件): 显示当前目录下的所有文件列表。此控件与上面两个控件一起可以组成一个文件系统，为用户提供文件处理能力，即提供给用户访问和处理驱动器、目录、文件的接口。

(16)  **Shape** (形状控件): 用于在窗体上绘制方框和圆等图形。

(17)  **Line** (直线控件): 用于窗体上绘制直线，其属性与 Shape 控件基本一致。

(18)  **Image** (图像控件): 类似于图形框控件，用于显示图形，但它只支持图形框控件的几个特性，需要的资源较少。

(19)  **Data** (数据控件): 该控件提供了访问存放在数据库中数据的一系列命令，它有很多属性，是数据库编程中很重要的一个数据访问控件。

(20)  **OLE Container** (OLE 容器控件): 容器控件是一个窗口，可以将其放在窗体中，用于放置其他应用程序中的文档。通过该控件，用户可以访问支持 OLE 的应用程序文档，如 Microsoft 的 Excel 文档。

当然还有许多可以添加到工具箱中的其他控件，这里就不全部罗列出来了。

在 VB 的 IDE 环境中还有工程管理器窗口，如图 1-5 所示。

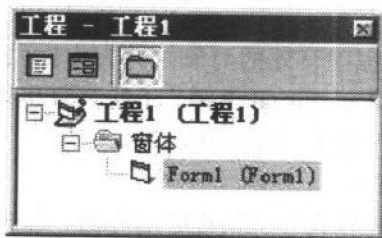


图 1-5 Visual Basic 的工程管理器窗口

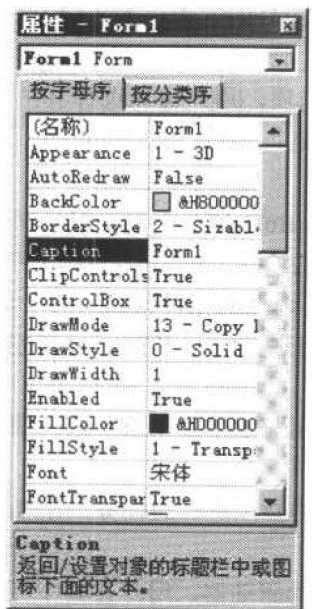


图 1-6 属性窗口

它用来列出当前工程中的窗体和模块。工程是指用于创建一个应用程序的文件集合。当你编写一个比较大的,有许多窗体模块的应用程序时,你就会喜欢上这个看上去不大的工程管理器窗口,因为它可以帮你方便、有效地管理整个应用程序中的各个窗体、模块,使你可以不费气力地在它们之间进行切换。

属性窗口如图 1-6 所示,相信它不久以后就会成为你编程过程中新结识的亲密无间的好朋友。

它用来列出选定窗体和控件的属性设置值。属性是指对象的特征,如大小、标题或颜色等。

在 Visual Basic 开发应用程序中窗体、控件等对象是用户接口的基本元素,图 1-6 属性窗口中列出的都是这些对象的表现形式,这跟我们现实生活中的概念是不冲突的,如现实生活中的一个对象——电视机,它有自己的颜色、型号、产地、数字还是模拟、产品编号等属性。

还有一个为我们开发程序提供便利的工具,就是对象浏览器,如图 1-7 所示。

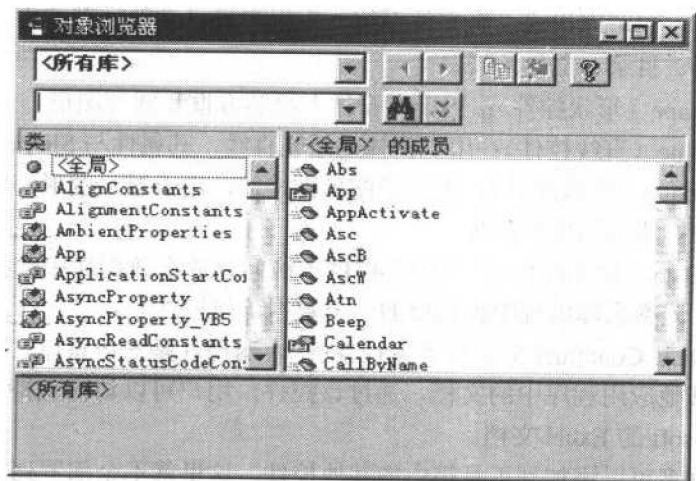


图 1-7 Visual Basic 的对象浏览器

在对象浏览器中列出工程中有效的对象,并为你提供了在程序代码中快速畅游的机会。可以使用对象浏览器浏览在 Visual Basic 中的对象和其他应用程序,查看对象有效的方法和属性,并将代码 COPY 进自己的应用程序。

窗体设计器如图 1-8 所示。

该设计器用来自定义窗口以设计应用程序的界面。在窗体中添加控件、图形和图片来创建所希望的程序外观。应用程序中每一个窗体都有自己的窗体设计器窗口。

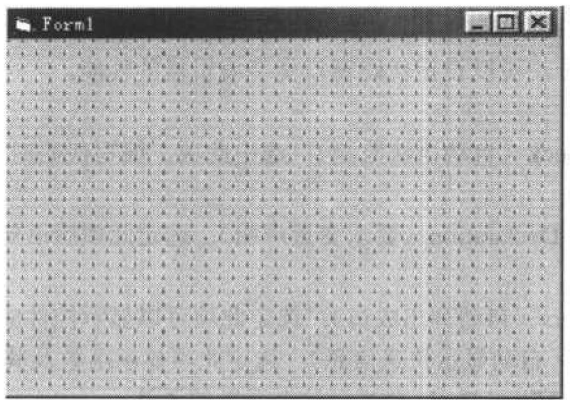


图 1-8 窗体设计器

代码编辑器窗口，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器窗口。

窗体布局窗口（Form Layout window）如图 1-9 所示。

窗体布局窗口允许使用表示屏幕的小图像来布置应用程序中各窗体的位置。



图 1-9 窗体布局窗口

立即、本地和监视窗口等附加窗口是为调试应用程序而提供的。它们只在 IDE 中运行应用程序时才有效。

在熟悉了 Visual Basic 的 IDE 环境后，你可以通过选择菜单栏里的“选项”，在弹出如图 1-10 所示的“选项”对话框里根据个人的爱好配置自己的工作环境。

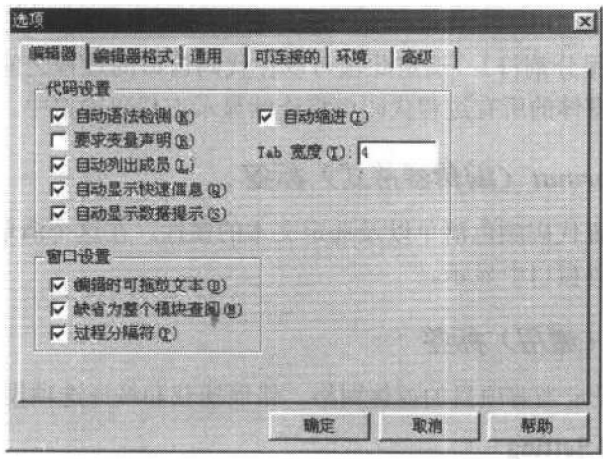


图 1-10 选项窗

下面我们对图 1-10 中的各个标签加以说明：

Editor（编辑器）标签

我们使用这个标签来设置代码和项目窗口。

✓ Code Setting (代码设置)

Code Setting 部分, 可以设定代码窗口文字编辑器的性能。

它有以下选项:

Auto Syntax Check (自动语法监测): 选中此项, 编写代码时如有错误按下【Enter】键会报警。

Require Variable Declaration (要求变量声明): 选定此项后语句 Option Explicit 将加入每个窗体和模块中。

Auto List Members (自动列出成员): 选定此项可以显示控件成员的弹出清单。

Auto Quick Info (自动显示快速信息): 选定此项显示函数及其参数信息, 如在程序中调用某个函数, 那么 Visual Basic 会显示这个函数的语法。

Auto Data Tips (自动显示数据提示): 选定此项后, 可以显示光标所在处变量的值。

Auto Indent (自动缩进): 选定此项可以跳表 (TAB) 第一行代码, 使以后各行代码均从跳表 (TAB) 位置开始。

✓ Tab Width (Tab 宽度)

Tab Width 部分, 在这个文本框中输入数字, 设置制表宽度, 有效值为 1~32。

✓ Windows Setting (窗口设置)

Windows Setting 部分, 用于指定 Code 编辑器的几个基本特性。

它有以下选项:

Drag-and-Drop Text Editing (编辑时可拖放文本): 选定此项后即可从 Code 代码窗口向立即窗口或监视窗口进行拖放操作。

Default to Full Module View (缺省为查看所有模块): 选定此项可以在代码窗口浏览过程清单。

Procedure (过程分割符): 选定此项可以在代码窗口的过程之间显示分割条, 如果不选定此项, 则一个窗体的所有过程代码都将连续显示在代码窗口中。

Editor Format (编辑器格式) 标签

该标签用于设置代码颜色清单以及选定文本的属性。在这个清单中可选择文本类型, 并指定其如何在代码窗口中显示。

General (通用) 标签

在这个标签中设定当前项目的窗体网格、错误捕获和编译选项设置。

✓ form Grid Setting

此部分可以设置指定放置控件窗体的网格特性, 包括网格的密度等。

它有以下选项:

Show Grid (显示网络): 选定此项指定显示窗体上的网格。

Grid (网格单位): 显示用于测量网格单位 (点数) 距离的单位。

Width (宽度)、Height (高度): 表示网格单元宽度和高度。

Align Control to Grid (对齐控件到网格): 用于调整控件大小和位置, 使其外边在网格线上。

Show ToolTips (显示工具提示): 选定此项显示工具条和工具框项目的工具提示。

Collapse Proj, Hides Windows (折叠工程、隐藏窗口): 选定此项则项目在工程资源管理器中重叠时隐藏窗口。

✓ **Error Trapping**

Error Trapping 部分, 可以指定在什么条件下, 应用程序中的错误产生运行错误报告或中断程序运行。

它有以下选项:

Break on All Errors (发生错误时中断): 虽然错误是由错误处理器处理, 当选定该模式任何错误都将导致 Visual Basic 进入断点状态。

Break on Class Module (在类模块内中断): 选定此项后, 当你的项目种类模块内没有错误处理器处理的错误, 将导致子项目进入断点状态。

Break on Unhandled (遇到未处理的错误时中断): 选定此项, 任何没有活动错误处理器的处理错误都将导致程序进入断点状态。

✓ **Compile**

Compile 部分, 用来设定 Visual Basic 何时以及如何在编译应用程序后开始执行应用程序。

它有以下选项:

Compile (请求时编译): 选定此项, 能使应用程序更快启动, 当开发人员选择编译时开始编译程序。

Background Compile (后台编译): 选定此项, 当你的应用程序没经过完全编译就开始执行时, Visual Basic 可利用闲置时间在后台进行项目编译。

Docking (可连接的) 标签

此标签是用来设定哪个窗口是可以连接的, 可连接窗口之间以及可连接窗口与主窗口之间是可以互相连接的。

Environment (环境) 标签

该标签包含了 Visual Basic IDE 环境中的各个属性。此标签中内容的改变将保存在 Registry 中, 每次启动 Visual Basic 时就会自动载入。

✓ **When Visual Basic Start**

When Visual Basic Start 部分, 用来设定在 Visual Basic 启动时发生的事项。

它有以下选项:

Prompt for project (提示创建工程): 选定此项, Visual Basic 启动时显示“新建项目”窗口。

Create default project (创建缺省工程): 选定此项, Visual Basic 启动时自动创建一个

缺省的 EXE 工程项目。

✓ **When a program starts**

When a program starts (启动程序时) 部分, 指定选择 Visual Basic IDE 主菜单中“运行”下“启动”菜单命令 (或按 F5 键), 启动应用程序时发生的事项。

它有以下选项:

Save Change (保存改变): 选定此项, 每次运行应用程序时将不作提示地自动保存对项目所做的改动。

Prompt To Save Changes (提示保存改变): 选定此项, 在运行应用程序时将提示是否保存对源代码的更改。

Don't Save Changes (不保存改变): 选定此项, Visual Basic 在运行程序时将既不自动保存改动也不提示保存。

Show Templates For (显示模板): 选定此项, 当你向应用程序中加入项目时工程菜单中将显示模板。

Templates Directory (模板目录): 在此项中输入存放模板文件的完整路径。



Advanced (高级) 标签

使用该标签可以设置如下项目:

Background Project Load (以后台方式加载工程): 选定此项, 可以设定转入方式为后台方式。

Notify when changing shared project items (当改变共享工程时提示): 可以在改变窗体和模块之类的共享程序项目并试图保存时, 让 Visual Basic 发出提示。

SDI Development Environment (SDI 开发环境): 选定此项, 可以在单文档接口 (SDI) 环境中开发程序。

以上就是对 Visual Basic 的 IDE 环境的介绍, 我们通过对选项菜单的熟练使用就可以马上为自己量身定制一个舒适的集成开发环境。

1.4 一章回顾

在这一章里, 我们主要介绍了 VB 6.0 的开发环境, 主要内容有:

- Visual Basic 的简单介绍, 让大家先认识 VB。
- 安装时应该注意的小问题, 比如联机注册, 安装数据访问应用程序的问题等, 帮助大家在自己的机器中安装好 VB。
- 详细介绍 VB 6.0 的集成开发环境, 使大家以最快的速度来掌握 VB 6.0 的开发环境、常用命令、基本控件并学会定制适合自己的开发环境。

在下一章中我们将向大家仔细地介绍 Visual Basic 的语法, 请紧跟我们的思路, 不久你就可以驾驭 VB 了。

第2章 Visual Basic 的语法

本章将介绍 Visual Basic 的语法，因为读者的水平参差不齐，建议刚出道的“好汉”们不要跳过本章，你将会在本章中对 Visual Basic 编程有一个感性的认识，为将来质的飞跃打好基础，“大侠”级别的可以跳过此章，当然最好再看看，共同探讨一下。

2.1 Visual Basic 应用程序的结构

一个应用程序实际上是指挥计算机完成用户所要完成任务的指令集。应用程序的结构是组织指令的形式，即指令存放的位置和执行顺序。虽然对于只有一行的代码来说，组织结构并不十分重要。但应用程序越复杂，代码就会越繁杂，这时组织或结构的重要作用就相当明显了。试想一下，如果允许应用程序以随机顺序去执行，那将会带来什么样的混乱。除了对应用程序的执行进行控制外，对于如何在应用程序中轻松查找特定指令，结构也起着不容忽视的作用。

Visual Basic 应用程序是面向对象的，应用程序的代码结构就是该程序在屏幕上可视的模型。这是第三代编程语言的典型特点。对象实质上是一种数据结构，包含了数据和对数据进行的操作，即数据和代码。以窗体为例，在屏幕上看到的窗体界面代表属性，这些属性决定了窗体的外观和内在特性。应用程序中所用的每个窗体都有一个包含其代码的相关窗体模块（文件扩展名为“.frm”），如图 2-1 所示。

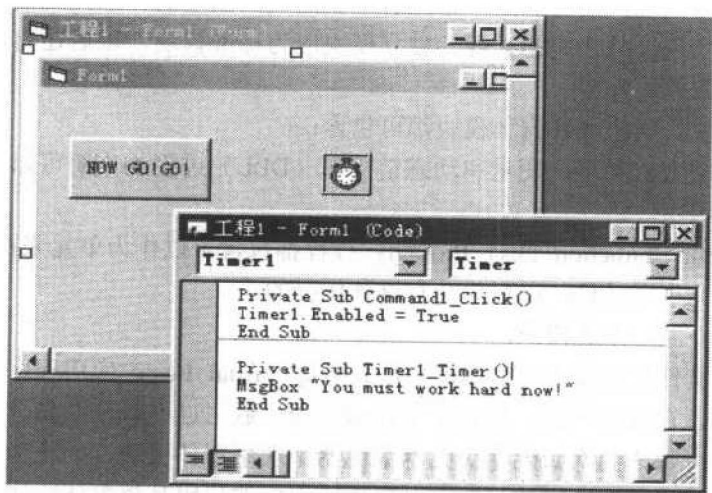


图 2-1 窗体及窗体相关的模块