

第 1 章 Visual Basic 基础

1.1 Visual Basic 概述

1.1.1 Visual Basic 简介

Visual Basic 源于 BASIC 语言。BASIC 语言是受到国内外千百万计算机爱好者欢迎的语言，自 1964 年问世以来，从实验室走向校园，从校园走向社会，从一个国度走向全世界，始终不衰。BASIC 是 **Beginners All Purpose Symbolic Interchange Code**（初学者通用符号代码）的缩写。与其他高级语言相比，它的语法规则相对简单，容易理解和掌握，且具有使用价值，被公认为是最理想的初学者语言。许多人由此入门成长为计算机应用人才。

BASIC 语言诞生以来，在广泛使用中不断发展，经历了四个主要阶段。第一代为早期 BASIC 也称为基本 BASIC。只有 17 条语句。第二代为微机 BASIC，以 GW-BASIC 为代表，是世界首富、微软公司创始人比尔·盖茨研制的。由于应用面广，微软公司由此起家。微机 BASIC 语言能处理数据文件，能制作图形、动画、声音，是功能丰富的实用的程序设计语言。第三代是 20 世纪 80 年代中期出现的结构化 BASIC 语言，以 QBASIC 为代表。以上三代都是 DOS 环境下的编程语言。第四代是 1991 年推出的 Windows 环境下的编程语言，就是本书所要介绍的 Visual Basic 语言（以下简称 VB），现在已经发展到 VB6.0 版本。

VB 在继承原 BASIC 语言简单易学的优点的同时，又吸收了近年来最先进、最优秀的编程技术，从而成为目前开发 Windows 应用程序最好、最快、最简单的开发工具之一。

实践证明，BASIC 较容易学习，BASIC 在不断发展，BASIC 拥有广大的学习者，BASIC 具有强大的生命力，BASIC 在计算机的普及中立下了汗马功劳。可以预言，BASIC 语言会随着计算机软、硬件的发展，继续充实、完善和优化自身，继续推出新版本，继续在计算机程序设计领域中发挥作用。

1.1.2 Visual Basic 功能特点

Windows 操作系统与 DOS 操作系统最主要的区别就是提供图形用户界面 GUI (Graphical User Interfaces)，Visual Basic 作为 Windows 环境下的编程语言适应 Windows 的特点，具有如下编程优势。

1. 可视化 Visual 的程序设计工具

在 DOS 环境下编程的高级语言，主要的工作是设计算法和编写程序。程序的各种功能

和显示界面都要用程序语句实现。而用 VB 开发应用程序，包括两部分工作：一是设计用户界面；二是编写程序代码。

VB 提供了一个“工具箱”，内放若干“控件”（例如制作按钮的控件、制作文本框的控件）。编程人员可以从工具箱中提取所需控件，放入窗体中合适的位置。也就是说，屏幕上的用户界面是用 VB 提供的可视化设计工具“控件”直接“画”出来的，而不是用程序“写”出来的。被认为最难办的界面设计，就这样被 VB 轻而易举地解决了。其实 VB 的界面设计也是由程序编写出来的，只是现在这些编程工作不由用户来做，而由 VB 系统来完成而已。

2. 面向对象的程序设计思想

面向对象的程序设计是伴随 Windows 图形界面的产生而产生的一种新的程序设计思想。所谓“对象”可以类比为现实生活中的可见“实体”。在程序设计当中，把用户界面上的每个实体，例如按钮、菜单、图片框以及窗体本身都称为“对象”，这些对象就是由可视化编程工具“控件”派生出来的。

不同的对象，在程序中所赋给它的功能是不同的。例如，用户界面上有两个命令按钮，一个用来处理用户输入的数据，另一个用来显示计算结果。两个按钮就是两个不同的对象，为了使这两个对象具有各自的功能，就要为这两个对象编写各自的代码，这种编程的思想和方法就是“面向对象的程序设计”。

3. 结构化程序设计语言

VB 是在 BASIC 和 Quick BASIC 语言的基础上发展起来的，保持了结构化程序设计的特点，接近自然语言和人类的逻辑思维方式，简单易用。

VB 是解释型语言，在输入代码的同时，解释系统将高级语言翻译成计算机可识别的机器指令，并判断每个语句的语法错误。应用程序设计好以后，可以编译生成可执行文件 (.exe)，脱离 VB，直接在 Windows 环境下运行。

4. 事件驱动的编程机制

VB 采用了事件驱动的编程机制，即对象与程序代码通过事件相联系。一个对象通常可以响应多个不同的事件，每个事件均能驱动一段程序（事件过程）的执行，从而完成该对象的某个功能。

例如，用户界面上有一个按钮。用户用鼠标点击该按钮，程序应完成“加法”运算。在这里，用户点击鼠标的动作会在该按钮对象上产生一个鼠标单击事件（Click 事件），于是 VB 系统就会自动调用执行命令按钮对象的 Click 事件过程，执行相应的程序代码。这就是事件驱动的功能。

VB 编程没有明显的主程序的概念。程序员所要做的就是在用户界面上设计各种对象，再面向不同的对象分别编写它们的事件过程。整个 VB 应用程序就是由这些彼此相互独立的事件过程构成。事件过程的执行与否以及执行的顺序取决于操作时用户所引发的事件。若用户未引发任何事件，则应用程序将处于等待状态。

5. 支持多种数据库系统访问

VB 提供了强大的数据库管理和存取操作的能力，利用数据控件可以访问的数据库系统有：Microsoft Access、Microsoft FoxPro、Paradox 以及任何支持开放数据库链接（ODBC）

标准的数据库。如 Microsoft SQL Server、Oracle、Sybase 等。也可以访问 Microsoft Excel、Lotus1-2-3 等多种电子表格。

另外，VB 还支持动态数据交换、对象的链接与嵌入等新型的编程技术，实现与其他 Windows 应用程序之间建立数据通讯。

1.1.3 Visual Basic 的版本

Microsoft 公司于 1991 年推出 Visual Basic 1.0 版，获得巨大成功。之后版本不断更新，1997 年推出的 5.0 版及 1998 年推出的 6.0 版均有中文版，大大方便了中国用户。

Visual Basic 6.0 包括 3 种版本，分别为学习版、专业版和企业版，这些版本是在相同的基础上建立起来的，故大多数应用程序可以在 3 种版本中通用。3 种版本适用于不同的用户。

学习版(Learning Edition) 是 VB 的基础版本。该版本包括了 VB 的所有内部控件(标准控件)、网络控件和数据绑定控件等。使用学习版可以十分容易地开发 Windows95/98/NT/Me/2000/XP 下的应用程序。

专业版(Professional Edition) 包括了学习版的所有功能。另外，增加了 ActiveX 控件、IIS 应用程序设计器、完整的数据工具和数据环境、数据对象以及 DHTML 设计器。该版本为专业编程人员提供了一套用于软件开发，功能完备的工具。

企业版(Enterprise Edition) 可以用来建立分布式的应用程序。它除了包括专业版的所有内容，还具有自动化管理器、部件管理器、数据库管理工具、Microsoft Visual SourceSafe 面向工程版的控制系统等。

版本越高，功能越全，占用系统资源越多，价格也越贵。用户应根据需要选用不同的版本。本书使用的是 Visual Basic 6.0 中文企业版，其内容可以用于专业版和学习版，所有的程序可以在专业版和学习版中运行。

1.2 Visual Basic 的运行环境、安装和启动

1.2.1 Visual Basic 的运行环境

在安装 VB 之前必须先确定自己的计算机能够满足最低安装要求。从 1991 年发布的 VB1.0 至今已有 6 个版本，不同版本的安装要求是不同的。例如 VB6.0 是一个 32 位的应用程序开发工具，其运行环境必须是 Windows95/98/NT/2000/XP。VB6.0 对系统的要求不高，为使它能以较流畅的方式工作，建议系统最低配置如下：

- 奔腾 133 以上 CPU。
- 至少 16(建议 32M 以上) 内存。
- 至少 80M 硬盘空间。
- 一个 CD-ROM 驱动器。

- VGA 以上分辨率的显示设备。

1.2.2 Visual Basic 的安装

VB 系统程序在发布时是经过压缩存储在光盘上的，使用前必须先将这些系统文件解压缩复制到硬盘上，这一过程通常称之为安装，其具体的解压和复制工作由光盘提供的安装程序（setup.exe）完成。以下以 VB6.0 企业版为例，介绍 VB 的安装。

（1）将 VB6.0 的安装光盘放入 CD-ROM 驱动器，并运行安装程序 SETUP.EXE。系统弹出“Visual Basic 6.0 中文企业版安装向导”对话框，如图 1.1 所示。

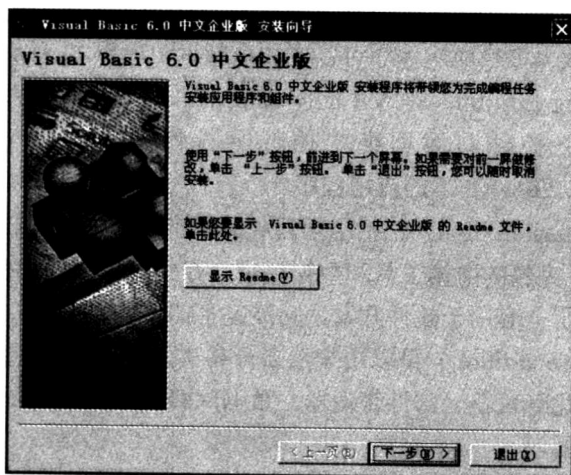


图 1.1 Visual Basic 6.0 中文企业版安装向导

（2）单击图 1.1 所示对话框中“下一步”按钮，则打开“最终用户许可协议”对话框，见图 1.2。用户必须点选“接受协议”，才可以点选“下一步”按钮进入下一界面。用户在该界面须输入产品的 ID 号，用户姓名和公司名称。

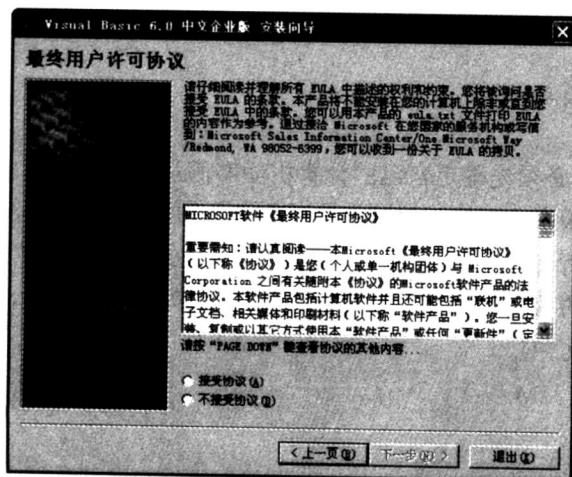


图 1.2 最终用户许可协议

(3) 单击“下一步”按钮，选择“安装 Visual Basic 6.0 中文企业版”见图 1.3 所示。

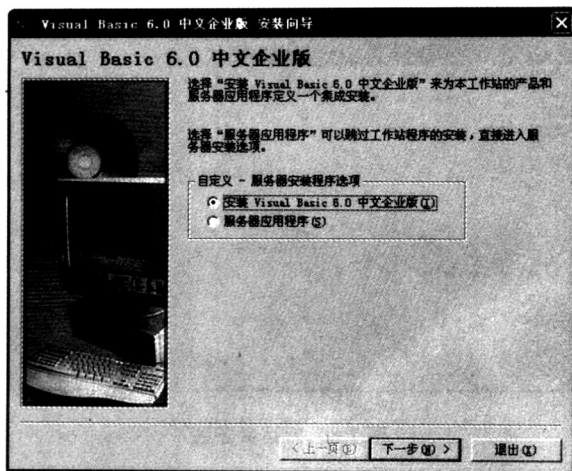


图 1.3 选择安装选项

(4) 单击“下一步”按钮，系统弹出安装类型选择对话框，如图 1.4 所示。这里有“典型安装”和“自定义安装”两个选择。“典型安装”将安装 VB 的一些常用组件，安装过程无须用户干预，建议初学者选择“典型安装”。安装 VB 的位置（路径）可以使用默认的文件夹，也可以单击“更改文件夹”按钮选择另外的路径。

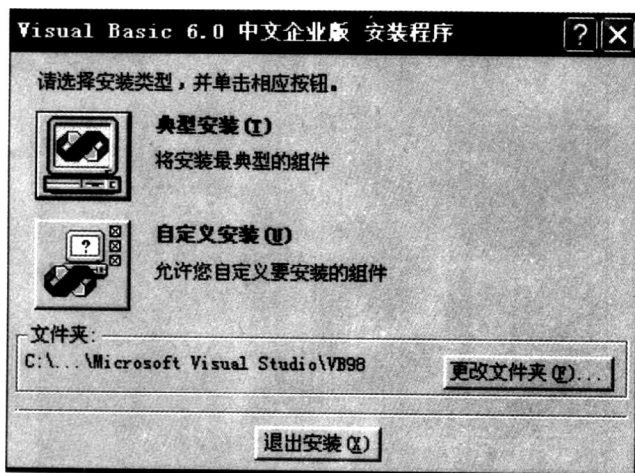


图 1.4 安装类型选择

若用户选择的是“自定义安装”，则弹出如图 1.5 所示的对话框。用户在这里可以根据需要有选择地安装某些组件。

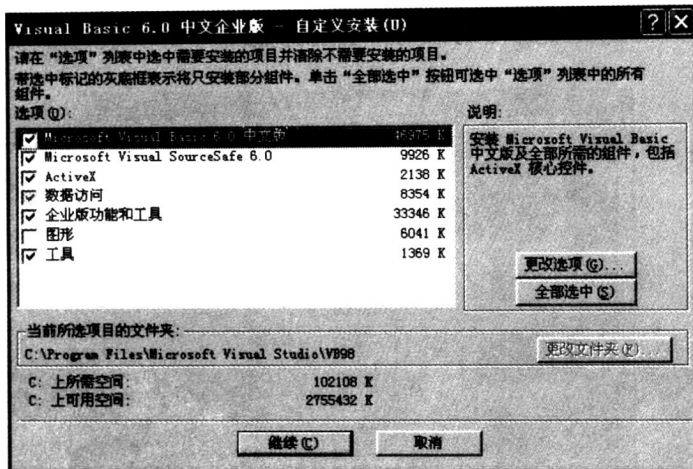


图 1.5 自定义安装

(5) 单击图 1.5 的“继续”按钮后, 安装程序将复制文件到计算机硬盘, 复制完毕重新启动计算机。

(6) 计算机重新启动后, 安装程序将自动打开“安装 MSDN”对话框, 如图 1.6 所示。MSDN 全称为 Microsoft Developer Network Library 包含约 1GB 的编程技术信息, 是编程人员的重要参考资料。VB6.0 的联机帮助文档只有在安装了 MSDN 后才可以使⤵用, 所以建议用户点选对话框中“安装 MSDN”前的选择框 再点击“下一步”。否则可以退出安装 MSDN。

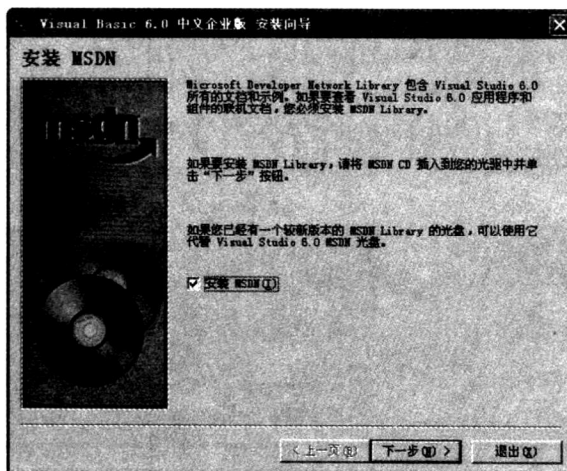


图 1.6 安装 MSDN

(7) 插入 MSDN 的第一张光盘到 CD-ROM, 单击图 1.6 中“下一步”按钮, 系统弹出安装类型选择对话框, 选择“自定义安装”选项后, 系统弹出“MSDN 自定义安装”对话框, 如图 1.7 所示。一般选中安装前三项, 再按“继续”按钮。以后再按照系统提示插入第二张 MSDN, 直至结束安装。

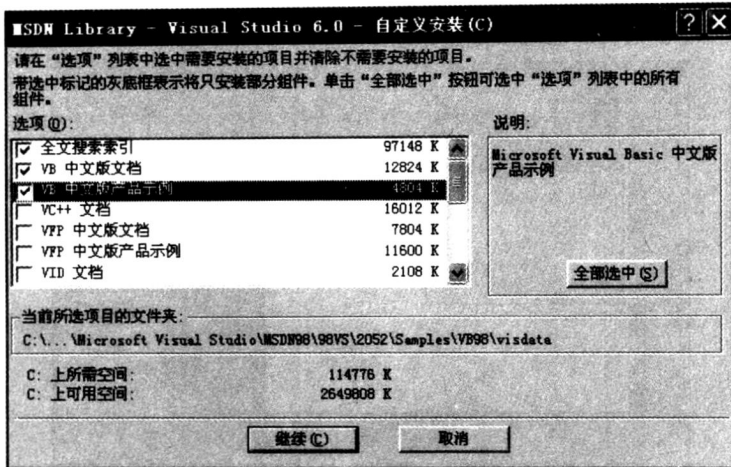


图 1.7 MSDN 自定义安装

1.2.3 Visual Basic 的启动与退出

VB 安装完成后, Windows 的开始菜单中就会出现启动 VB 的菜单选项。点击“ Microsoft Visual Basic 6.0 中文版 / Microsoft Visual Basic 6.0 中文版 ”即可启动 VB6.0。当然, 通过“ 我的电脑 ”找到 VB6.EXE , 双击该文件名, 也可启动 VB6.0。VB6.0 启动后, 会弹出“ 新建工程 ”对话框, 如图 1.8 所示。

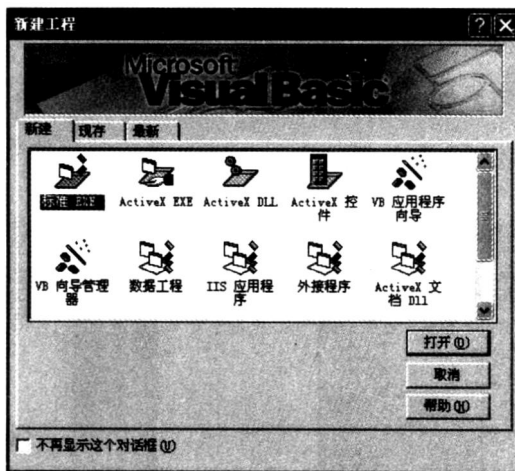



图 1.8 新建工程

双击新建选择项卡中的缺省选项“ 标准 EXE ”或直接点击“ 打开 ”按钮, 即进入 VB 的集成开发环境, 如图 1.9 所示。这个集成开发环境集中了界面设计、程序设计、编辑、编译、调试等各种功能, 我们的应用程序开发工作就在这里进行。如果需要编辑一个已经存在的 VB 应用程序, 可以执行“ 文件 ”菜单的“ 打开工程 ”命令 (快捷键 Ctrl+O), 也可以直接单击工具条上的  按钮。

应该注意，VB 是开发 Windows 系统的应用程序，因此不能从 DOS 系统中启动。

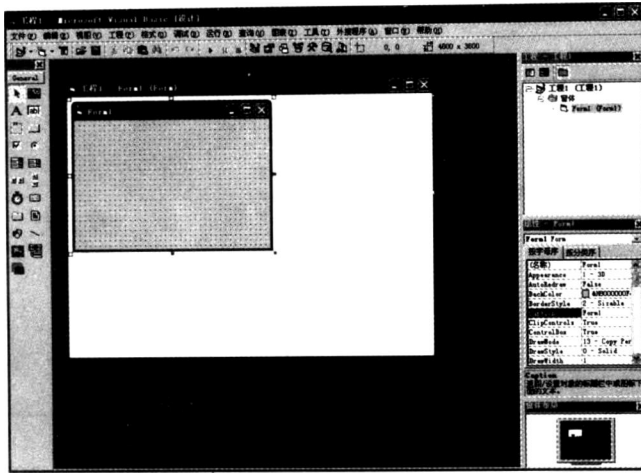



图 1.9 VB 集成开发环境

单击图 1.9 右上角的  关闭按钮或选择文件菜单中的“退出”命令（快捷键 Alt+Q），均可退出 VB。退出前，VB 系统会根据当前文件是否已存盘，提示用户保存当前文件。

1.3 Visual Basic 快速入门

1.3.1 认识 Visual Basic 开发环境

VB 集成开发环境中各部分组成如图 1.10 所示。

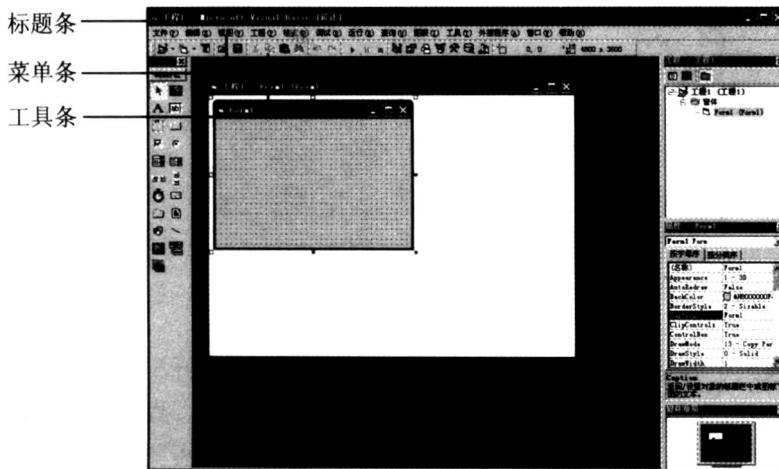


图 1.10 VB 集成开发环境的组成

1. 标题条

位于窗口的顶部，它用来显示窗口的标题，如工程 1。标题最后的方括号内指出目前处在设计模式、运行模式或调试（break）模式。图 1.10 表示当前处在 VB 开发环境，正在设计的是工程 1，是设计模式。
















2. 菜单条

位于标题条下方，包含文件（F）、编辑（E）等 13 个菜单项。每个菜单项后面有一个带下划线的大写英文字，是快捷标识。如“文件”后面的（F），表示按 Alt+F 即可打开菜单上的命令，即可执行相应操作，有些常用命令后也有快捷键供选用。在下文中为叙述简洁，用“文件 / 打开工程”表示选择“文件”菜单下的“打开工程”命令。








3. 工具条及坐标显示区

位于菜单条下方，它以图标按钮的形式提供了常用的菜单命令。例如用鼠标点击第 4 个标有打开图标的按钮，就相当于选择了“文件”菜单的“打开工程”命令。所有按钮的功能见表 1.1。

表 1.1 标准工具按钮

图标	名称与功能	快捷键
	添加 Standard EXE 工程。用来添加新的工程到工作组中。单击右边的箭头，将弹出一个下拉菜单，可以从中选择需要添加的工程类型	
	添加窗体。用来添加新的窗体到工程中，单击其右边的箭头，将弹出一个下拉菜单，可以从中选择需要添加的窗体类型	
	菜单编辑器。显示菜单编辑器对话框	Ctrl+E
	打开工程。用于打开已有的工程文件	Ctrl+O
	保存工程。用于保存已有的工程文件	
	剪切	Ctrl+X
	复制	Ctrl+C
	粘贴	Ctrl+V
	查找	Ctrl+F
	撤销键入	Ctrl+Z
	重复键入	
	启动。开始运行当前工程	F5
	中断。暂时中断当前工程的运行	Ctrl+Break
	结束。结束当前工程的运行	
	工程资源管理器。打开工程资源管理器窗口	Ctrl+R
	属性窗口。打开属性窗口	F4

续表 1.1

图标	名称与功能	快捷键
	窗体布局窗口。打开窗体布局窗口	
	对象浏览器。打开对象浏览器对话框	F2
	工具箱。打开工具箱窗口	
	数据视图窗口。打开数据视图窗口	
	可视化部件管理器。打开可视化部件管理器	
	0, 0  4800 x 3600	数字显示区。显示当前对象的位置与大小

表中数字显示区包括两部分，左数字区显示的是对象的坐标位置[窗体的左上角为坐标原点，即(0, 0)点]；右数字区显示的是对象的高度和宽度，即对象的大小。数字的单位为 Twip，是一种与屏幕分辨率无关的计量单位。1 英寸等于 1440Twip，即无论在什么屏幕上，一条长 1440 Twip 的直线，打印出来都是 1 英寸。这种计量单位可以确保在不同的屏幕上都能保持正确的相对位置或比例关系。

4. 控件工具箱

位于窗体的左侧，提供了最常用的界面设计工具（可视化控件）。这些工具以图标的形式排列在工具箱中。程序员可以用这些工具在窗体上构造出所需的应用程序界面。各工具的名称见图 1.11。

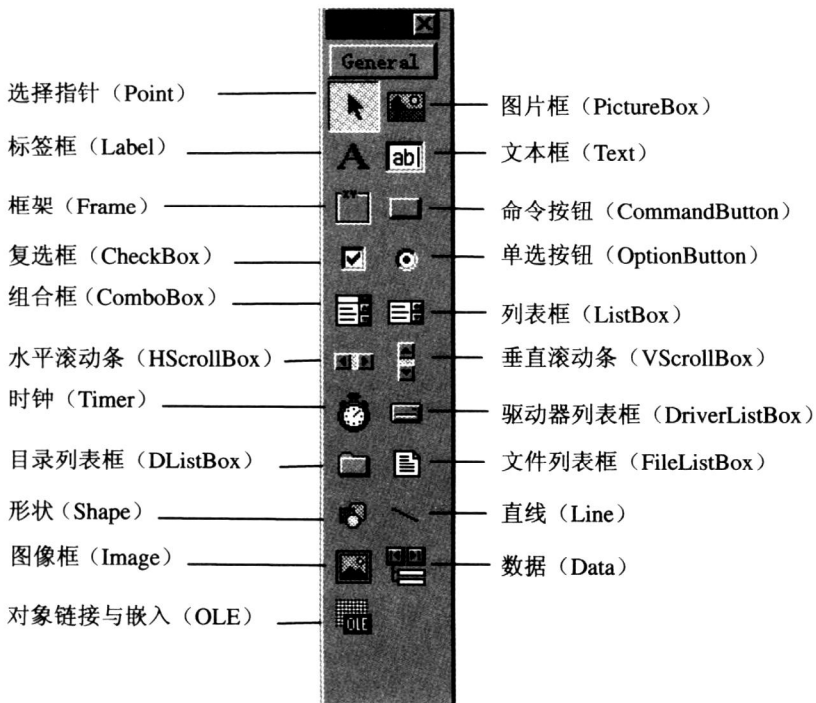


图 1.11 控件工具箱

5. 窗体窗口

位于屏幕正中，程序员在这里设计用户所需的应用程序界面。窗体的缺省状态是其上布满网格，是为了便于安排窗体上对象的大小及位置。可以通过点选菜单命令“工具 / 选项 / “通用”卡 / 窗体网格设置”取消或改变网格大小。

6. 工程窗口

位于屏幕的右上方，它列出当前应用程序所包含的文件清单，所以也称为工程资源窗口。一个VB应用程序可以包含多类文件：扩展名为 frm 的窗体文件、扩展名为 vbp 的工程文件、扩展名为 vbg 的工程组文件、扩展名为 bas 的程序模块文件、扩展名为 cls 的类模块文件、扩展名为 frx 的二进制文件、扩展名为 res 的资源文件等。

工程窗口标题条下有 3 个按钮，它们的功能见图 1.12。同 Windows 系统的资源管理器一样，目录可以被折叠（左边方框中为“+”号），也可以被展开（左边方框中为“-”号）。

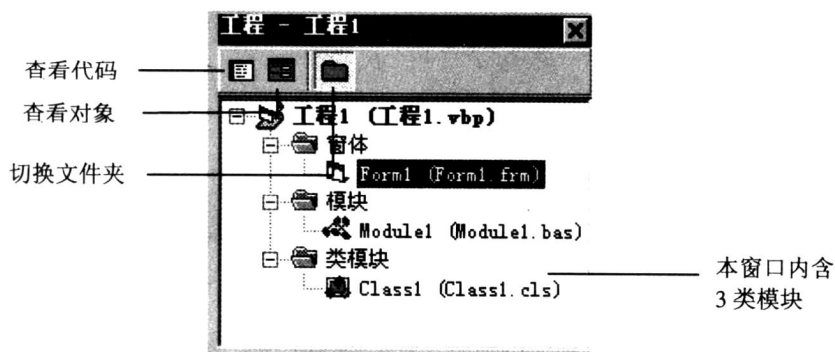


图 1.12 工程窗口

7. 属性窗口

位于项目窗口下方。属性窗口中有一个对象下拉列表框，其内列出当前窗口中的所有对象。从中选中一个对象，就激活了当前窗口中该对象，位于其下的列表就显示该激活对象的属性。见图 1.13。

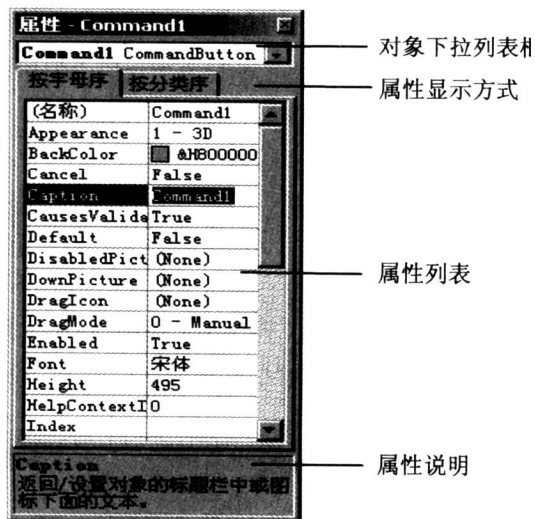


图 1.13 属性窗口

8. 窗体布局窗口

位于属性窗口的下方。可以在该窗口设置当前窗体在屏幕中显示时的初始位置。

1.3.2 入门例题

于此，我们通过一个最简单的 VB 应用程序，引导初学者对 VB 编程有一个概括性的了解。

【例 VB1-1】设计一个程序，其上有一个命令按钮。当程序运行时，在运行中若用鼠标单击按钮，在窗体上就会显示出“Visual Basic 欢迎您！”一行字。运行界面如图 1.14 所示。

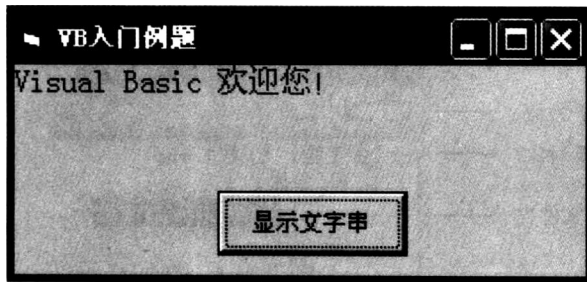


图 1.14 VB1-1 运行界面

一般说来，设计一个 VB 应用程序，需要以下 4 步：设计用户界面，设计对象属性，编写程序代码，保存与调试。这 4 步的具体操作如下：

步骤一：设计用户界面。

设计一个用户使用程序时的界面，用户输入或输出的信息都在这个界面中进行。界面设计的原则是让用户感到方便美观。

本例用户界面中只需要一个命令按钮。点击工具箱中的命令按钮，鼠标指针变为十字形，按住鼠标左键，在窗体中拖出一个按钮；或者双击工具箱中的按钮控件，则在窗体正中会出现一个命令按钮对象，再用鼠标将该按钮拖到窗体中合适位置。

通常，我们把工具箱中的工具称为“控件”，把用“控件”在窗体中“画”出的图形称为“对象”，窗体中可以画出很多对象，但只有一个对象是当前激活的对象，其四周有 8 个蓝色小方块，称为控制柄，用鼠标拖动控制柄可以改变对象大小。

步骤二：设置对象属性（特征）。

激活属性窗口，打开其内的对象下拉列表框（见图 1.13），可以看到当前窗体共有两个对象，分别是窗体（Form1）、命令按钮（Command1），可见窗体本身是一个不用工具箱中的控件生成的对象。

激活窗体（用鼠标点击窗体本身，使之四周出现控制柄。或选择属性窗口中对象下拉列表中的 Form1），设置窗体的属性如图 1.15 所示。



图 1.15 设置窗体属性

请关注图 1.15 中下述三个属性：


Caption (标题): VB 入门例题；

名称(Name): Form1 (缺省)；

Font (字体大小):小四；

我们修改了其中的 Caption 与 Font 属性，窗体的其他属性可以不修改，使用默认值。

在窗体的 Caption 属性中输入的文字串“VB 入门例题”，显示在窗口的标题条上，是给用户看的。窗体的名称 Form1 是该窗体的唯一标识，一个工程中不能有同名的窗体，这个名称用来在程序代码中区别不同的对象。名称属性与 Caption 属性的区别是：前者作为标识，是程序员在编写代码时使用的，用户看不到；后者显示在用户界面上，是给用户看的。

点击属性窗口的 Font 属性，右边弹出有 3 个黑点的小按钮 ，点击该小按钮，会弹出一个字体对话框，见图 1.16。可以在其中设置窗体中显示字体的字体名、字体样式、字体大小等。本例只修改字体的大小为“小四”，其他属性用默认值。

接着设置命令按钮的属性。当然，也先得激活按钮，方法同激活窗体一样。在设置按钮属性前，窗体中按钮上的文字为“Command1”。显然，让用户看这个英文名字实在让人不知所云，不可能知道这个按钮究竟是提供什么功能的。所以我们应该把它改掉。由于我们要告诉用户，按这个按钮可以显示文字串，所以修改命令按钮的 Caption 属性为“显示文字串”，修改后，立即可以看到窗体上该命令按钮上的“Command1”已经改为“显示文字串”。

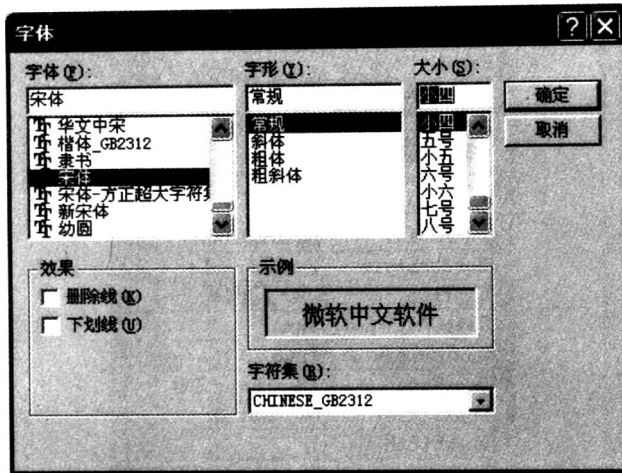


图 1.16 设置字体

缺省命令按钮的名称属性与 Caption 属性是一样的。例如该命令按钮两个属性值均为 Command1。这个名字只说明了这是该窗体中的一个命令按钮 (Command)，其后的序号是“1”，是系统按照制作先后顺序自动给出的。在程序设计过程当中，一般应该给按钮一个“见名知义”的名称，以便于编写代码时使用。修改按钮名称为 CmdDisplay。虽然 VB 是不区分大小写的语言，但仍建议程序设计者按 VB 的规范命名对象。这个规范是前 3 个字母 (前缀) 是对象类型的缩写，且第一个字母大写 (例如 Cmd 表示命令按钮)；前缀后面是该对象的功能描述，首字母也大写 (如 Display)。两者之间不要空格，否则系统会误认为这是两个对象或两个变量。在下文中，我们会逐步介绍各类对象前缀的写法。

步骤三：编写程序代码。

通过步骤一和步骤二，用户已经可以看到用户界面上的按钮，及按钮上的提示信息 (即为标题 Caption)，但是这时若用户用鼠标点击该命令按钮，应用程序不会有任何反应。所以还要为命令按钮编写一段程序。

编写程序代码要在“程序代码窗口”中进行。图 1.10 是设计用户界面的窗体，进入程序代码窗口有如下 4 种方法：

- 双击当前窗体或窗体上某个对象。
- 单击工程窗口的“查看代码”按钮，见图 1.12。单击工程窗口的“查看对象”按钮又会回到窗体设计窗口。
- 在菜单条中选择“视图 / 代码窗口”。
- 按 F7 功能键。

代码窗口如图 1.17 所示，我们在左上方的对象下拉列表框中选“CmdDisplay”，在右上方的过程下拉列表框中选“Click”。这时代码窗口将自动出现代码：

```
Private Sub CmdDisplay_Click()
|
End Sub
```

输入光标在两行代码中间的空行上闪烁，提示程序员在光标处输入代码。本例只需要一行代码：Print "Visual Basic 欢迎您！"（用西文双撇号，不能用中文双引号）

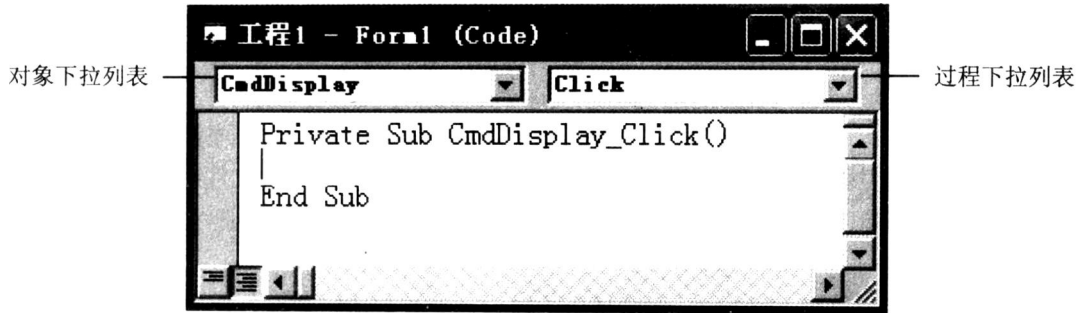


图 1.17 程序代码窗口

其中，关键字 Private（私有）表示该过程只能在本窗体文件中被调用，应用程序中的其他窗体或模块不可以调用它。关键字 Sub 是子程序的标志。CmdDisplay_Click() 是过程名，它由两部分组成：对象名和事件名，之间用下划线连接。End Sub 表示过程结束。

步骤四：保存与调试。

通过上述 3 个步骤，应用程序基本制作好了，这时应先保存文件，以防运行时因意外错误情况引起“死机”而造成文件的丢失。

VB 应用程序的保存与通常的文件存盘有些不同。一个 VB 应用程序是一个工程或一个工程组。VB 工程是由与该工程相关的若干文件组成。例如：扩展名为 frm 的窗体文件，扩展名为 vbp 的工程文件，扩展名为 frx 的二进制文件等。应用程序越复杂，所包含的文件越多，但一个应用程序至少要包含 frm 文件和 vbp 文件。

选择菜单“文件 / 保存工程”或选择工具条上的“保存工程”按钮，将弹出保存窗口如图 1.18 所示。

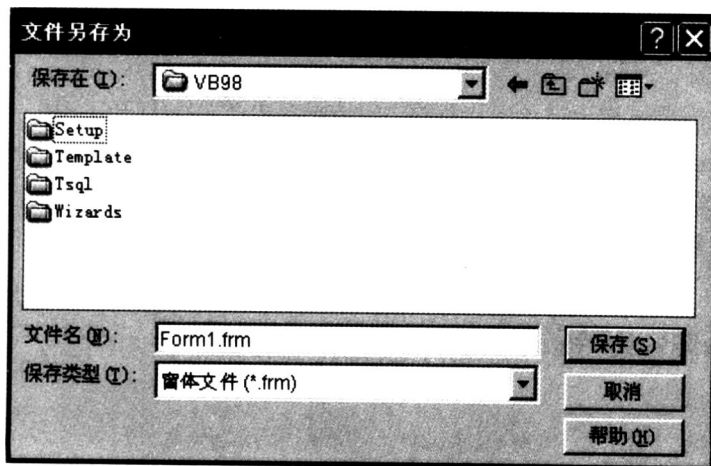






图 1.18 保存窗口

注意，这时系统已给窗体文件命名一个缺省的名字：Form1.frm，该缺省名表示该工程第一个窗体的窗体文件。建议不要使用这个名字，换成“见名知义”的名字。例如本例，名字起为 VB1-1.frm，表示这是 VB 第一章中第一个例题的窗体文件。保存完窗体，按“保存”按钮后，系统接着会再次弹出另一个保存窗口，这时文件名显示的缺省名为：工程 1.vbp。把这个名字改为 VB1-1.vbp，这样在资源管理器目录中我们很容易识别 VB1-1.frm 与 VB1-1.vbp 是一个工程中的文件，它们互相依存，缺少哪一个，应用程序都不能正常运行。也就是说，当我们移动这个应用程序在硬盘中的位置时，其中的所有文件必须一起移动。如果该工程中的所有文件都是同名的（仅扩展名不同），在资源管理器中把它们按名称排序，就很容易把它们一个不漏的选中了。

保存好文件就可以试运行了。程序是否能以设计者预想的方式运行呢？除非如本例这种非常简单的例子，一般情况下都不可能一次成功，需要反复试运行，反复修改，这个过程叫做调试。

如 1.1.2 节所述，VB 具有一个非常友好的集成开发环境，可以集界面设计、代码编写、运行调试于一体。在该集成环境中进入运行模式有两种方法：

- 通过菜单命令：“运行 / 启动”或“运行 / 全编译执行”
- 通过工具条的启动按钮，在运行模式下工具条中的暂停按钮与停止按钮可以用来控制程序的运行。

若程序运行不正常，就要返回到设计模式中修改。点停止按钮，程序会自动返回到设计模式。修改后应再次存盘（点按工具条上“保存工程”按钮），但这次存盘不会弹出为文件命名的窗口，系统会自动按原路径原文件名重新存储该工程内的所有文件，例如 frm 文件及 vbp 文件。

当确认程序运行正常，可以选择菜单命令“文件 / 生成 VB1-1.exe”（exe 文件可以脱离 VB 环境运行），系统缺省将*.vbp 的文件名作为可执行文件的文件名，本例工程文件名是 VB1-1.vbp，所以“文件”菜单下的生成 exe 文件名已变成“生成 VB1-1.exe”。程序员可以在后续弹出的窗口中修改这个名字，但建议不要修改，仍是为了能方便的找到与该应用程序相关的所有文件。

如果日后需要再次调试该应用程序 应选择菜单“文件 / 打开工程”；打开工程窗口显示，只允许打开 vbp 文件，可见打开工程要通过 vbp 文件，vbp 文件会自动调用它所需要的其他文件。有些初学者在资源管理器中双击 frm 文件，某些时候也能正常运行，但当窗体多于一个或遇到其他复杂应用程序时就会出错。

1.4 Visual Basic 帮助系统

1.4.1 上下文相关帮助

上下文相关帮助是不用程序员查找，在 VB 集成开发环境中哪里遇到了问题，系统就会

智能地弹出相关帮助，例如将光标放在代码窗口的 Sub 上，再按 F1 键，就会弹出 Sub 帮助信息窗口，如图 1.19 所示。



图 1.19 选择上下文帮助的目标

可以使用上下文相关帮助的部分有：

- VB 中的每个窗口（包括属性窗口、代码窗口等）
- 工具箱中的控件。
- 窗体或其中的对象。
- 属性窗口中的属性。
- VB 关键词（如声明、函数、属性、方法、事件和特殊对象）。
- 错误信息。

获取上下文相关帮助的方法是一样的：激活目标，然后按 F1 键。

1.4.2 使用 MSDN Library 在线帮助

VB 向用户提供了很好的在线帮助和自学功能，对广大读者学习和使用 VB 带来了极大的方便。使用 VB 的帮助系统，不仅可以引导初学者入门，而且给出了大量的详细信息，可以帮助各种层次的程序员完成程序的设计。它是一本集程序设计指南、用户使用手册以及函数库于一体的电子词典。所以学会使用帮助系统是十分重要的。

VB 的帮助系统 MSDN Library 全称为 Microsoft Developer Network Library 通过菜单命令“帮助/内容”或“帮助/索引”或“帮助/搜索”都可以打开 MSDN 窗口，见图 1.19 所示。

MSDN 窗口上有左右两个窗格，左边是定位窗格，右边是主题窗格。定位窗格中含“目录”、“索引”、“搜索”和“书签”4 张选项卡。

“目录”选项卡对不同的帮助主题进行了多层次的分类，用户可以通过鼠标点击目录前