



信息素养文库 · 高等学校信息技术系列课程规划教材

# Visual Basic

## 程序设计教程

◎ 主 编 许 洋 聂黎生 周晓云

◎ 副主编 田慧珍 徐 建 耿夫利



南京大学出版社



# Visual Basic 程序设计教程

◎ 主 编 许 洋 聂黎生 周晓云  
◎ 副主编 田慧珍 徐 建 耿夫利



【微信扫码】  
本书导学，领你入门



## 内容简介

高校非计算机专业学生的教学计划中用于 Visual Basic 的学时有限,传统教材的内容过于详尽,面面俱到,学生学习时很难把握主次。本书力图用简明的语言、典型的例题、通俗的解释将枯燥乏味的理论知识传授给学生,循序渐进地介绍 Visual Basic 的概念、语法基础和应用实例,并辅之以图表说明,实用性强,学生容易上手。

本书每章后面都附加了丰富的习题,大多取材于等考真题,既有复习的作用,又有备考的效果。

与本书配套出版的《Visual Basic 上机实验与学习指导》主要包括考试真题和上机实践指导两部分内容。由浅入深地配合主教材编排内容,其中有的题目给出了实验步骤、程序代码,有的只提供了设计思路和提示,有的只有题目要求,逐步培养学生独立分析和解决问题的能力。

## 图书在版编目(CIP)数据

Visual Basic 程序设计教程 / 许洋, 聂黎生, 周晓云主编. — 南京: 南京大学出版社, 2018.1

(信息素养文库)

高等学校信息技术系列课程规划教材

ISBN 978-7-305-19753-6

I. ①V… II. ①许… ②聂… ③周… III. ①BASIC 语言—程序设计—高等学校—教材 IV. ①TP312.8

中国版本图书馆 CIP 数据核字(2017)第 317743 号

出版发行 南京大学出版社

社 址 南京市汉口路 22 号 邮 编 210093

出版人 金鑫荣

丛 书 名 信息素养文库·高等院校信息技术课程精选规划教材

书 名 Visual Basic 程序设计教程

主 编 许 洋 聂黎生 周晓云

责任编辑 王秉华 王南雁 编辑热线 025-83597482

照 排 南京南琳图文制作有限公司

印 刷 丹阳市兴华印刷厂

开 本 787×1092 1/16 印张 15 字数 350 千

版 次 2018 年 1 月第 1 版 2018 年 1 月第 1 次印刷

ISBN 978-7-305-19753-6

定 价 37.80 元

网址: <http://www.njupco.com>

官方微博: <http://weibo.com/njupco>

官方微信号: njupress

销售咨询热线: (025) 83594756

\* 版权所有,侵权必究

\* 凡购买南大版图书,如有印装质量问题,请与所购图书销售部门联系调换

# 前 言

Visual Basic 作为一种结构化、模块化、面向对象、包含协助开发环境,以事件驱动为机制的可视化程序设计语言,具有功能强大、易学易用的特点,成为许多程序设计初学者首选的编程语言。人们可以用它轻松快速地建立应用程序。

许多单位、部门把掌握一定的计算机知识和应用技能作为人员聘用、职务晋升、职称评定、上岗资格的重要依据之一。全国计算机等级考试(NCRE)为此提供了一个统一、客观、公正的标准。编者根据十多年 Visual Basic 教学经验和《全国计算机等级考试二级 Visual Basic 考试大纲》的要求编写了本书,旨在提高学生计算机应用的理论水平和实际操作能力。本书既可作为二级 Visual Basic 科目的培训教材与自学用书,也可作为学习 Visual Basic 的参考书。书中以 Visual Basic 语言为工具介绍程序设计的基本思想和基本方法,为读者将来学习其他编程语言打下良好的基础。

本书的编写团队由江苏师范大学智慧教育学院基础部多年从事 Visual Basic 教学的一线教师组成,经验丰富。书中一、二章由聂黎生老师编写,三、四章由团队共同编写,五、六章由田慧珍老师编写,七、八、九章由许洋老师编写,徐建老师参与编写了三、四章。周晓云教授在百忙之中审阅了全部书稿,并提出了许多宝贵的意见和建议。在此对她的辛勤付出表示衷心的感谢!

本书还配套有不少网络资源,内容包括导学、习题解答、在线练习、其他相关资源等,覆盖相关章节,能够让学习者随时随地用手机观看。这些网络资源以二维码的形式在书中呈现,无需下载与注册,只需用微信扫描即可查阅。

由于作者水平有限,书中错误和缺点在所难免,恳请读者批评指正。

编 者

2017 年 12 月

# 目 录

## 第 1 章 Visual Basic 程序开发环境

|                                       |    |
|---------------------------------------|----|
| .....                                 | 1  |
| 1.1 可视化与事件驱动编程机制 ...                  | 1  |
| 1.1.1 编程的可视化 .....                    | 1  |
| 1.1.2 事件驱动的编程机制 .....                 | 1  |
| 1.2 Visual Basic 的启动与退出 .....         | 2  |
| 1.2.1 VB 6.0 的启动 .....                | 2  |
| 1.2.2 VB 6.0 的退出 .....                | 3  |
| 1.3 Visual Basic 集成开发环境(IDE)的组成 ..... | 3  |
| 1.3.1 VB 的三种工作模式 .....                | 3  |
| 1.3.2 VB 集成开发环境介绍 .....               | 4  |
| 1.4 对象及其操作 .....                      | 7  |
| 1.4.1 对象(Object) .....                | 7  |
| 1.4.2 属性(Property) .....              | 7  |
| 1.4.3 事件(Event) .....                 | 8  |
| 1.4.4 方法(Method) .....                | 8  |
| 1.4.5 属性、方法和事件之间的关系 .....             | 8  |
| 1.5 程序设计的一般步骤 .....                   | 9  |
| 1.5.1 创建用户界面 .....                    | 9  |
| 1.5.2 设置对象属性 .....                    | 10 |
| 1.5.3 编写事件代码 .....                    | 11 |
| 1.5.4 保存程序文件 .....                    | 12 |
| 1.5.5 运行及调试程序 .....                   | 12 |
| 1.5.6 生成可执行程序 .....                   | 12 |
| 本章习题 .....                            | 13 |

## 第 2 章 创建用户界面

|                |    |
|----------------|----|
| 2.1 创建窗体 ..... | 14 |
|----------------|----|

|                            |    |
|----------------------------|----|
| 2.1.1 窗体的结构、常用属性和事件 .....  | 14 |
| 2.1.2 定制窗体的属性 .....        | 17 |
| 2.1.3 窗体的装载、卸载和显示、隐藏 ..... | 17 |
| 2.1.4 Print 方法 .....       | 18 |
| 2.1.5 与 Print 有关的函数 .....  | 20 |
| 2.1.6 Move 方法 .....        | 20 |
| 2.1.7 Cls 方法 .....         | 21 |
| 2.1.8 格式输出函数 Format .....  | 21 |
| 2.2 VB 常用控件 .....          | 22 |
| 2.2.1 概述 .....             | 22 |
| 2.2.2 常用控件 .....           | 23 |
| 2.3 制作菜单 .....             | 37 |
| 2.3.1 菜单概述 .....           | 37 |
| 2.3.2 用菜单编辑器建立菜单 ...       | 39 |
| 2.3.3 创建弹出式菜单 .....        | 40 |
| 2.4 多窗体和多文档界面 .....        | 42 |
| 2.4.1 多窗体设计 .....          | 43 |
| 2.4.2 多窗体程序设计示例 .....      | 44 |
| 本章习题 .....                 | 47 |

## 第 3 章 Visual Basic 程序设计基础

|                                     |    |
|-------------------------------------|----|
| .....                               | 50 |
| 3.1 过程与模块 .....                     | 50 |
| 3.1.1 过程 .....                      | 50 |
| 3.1.2 模块 .....                      | 51 |
| 3.2 Visual Basic 程序的书写规范和常用语句 ..... | 52 |
| 3.2.1 书写规范 .....                    | 52 |
| 3.2.2 常用语句 .....                    | 52 |
| 3.3 数据类型 .....                      | 54 |

|                                        |     |                                 |     |
|----------------------------------------|-----|---------------------------------|-----|
| 3.3.1 基本数据类型 .....                     | 54  | 5.2.1 动态数组的定义 .....             | 113 |
| 3.3.2 用户定义的数据类型 .....                  | 56  | 5.2.2 Erase 语句 .....            | 115 |
| 3.4 常量和变量 .....                        | 57  | 5.3 数组的基本操作 .....               | 116 |
| 3.4.1 常量 .....                         | 58  | 5.3.1 数组元素的引用 .....             | 116 |
| 3.4.2 变量 .....                         | 60  | 5.3.2 数组元素的赋值 .....             | 117 |
| 3.5 运算符与表达式 .....                      | 64  | 5.3.3 For Each ...Next 语句 ..... | 123 |
| 3.5.1 算术运算符 .....                      | 64  | 5.4 控件数组 .....                  | 125 |
| 3.5.2 关系运算符与逻辑运算符<br>.....             | 66  | 5.4.1 基本概念 .....                | 125 |
| 3.5.3 表达式的执行顺序 .....                   | 69  | 5.4.2 建立控件数组 .....              | 125 |
| 3.6 常用内部函数 .....                       | 70  | 5.4.3 使用控件数组 .....              | 126 |
| 3.6.1 数学函数 .....                       | 70  | 5.5 程序设计举例 .....                | 127 |
| 3.6.2 转换函数 .....                       | 72  | 本章习题 .....                      | 135 |
| 3.6.3 字符串函数 .....                      | 73  | <b>第 6 章 过 程</b> .....          | 138 |
| 3.6.4 日期和时间函数 .....                    | 76  | 6.1 事件过程 .....                  | 138 |
| 3.6.5 InputBox 函数与 MsgBox 函数<br>.....  | 77  | 6.2 Sub 子过程 .....               | 140 |
| 本章习题 .....                             | 82  | 6.2.1 建立通用 Sub 子过程 .....        | 140 |
| <b>第 4 章 选择结构与循环结构</b> ..              | 83  | 6.2.2 调用 Sub 过程 .....           | 142 |
| 4.1 选择结构 .....                         | 83  | 6.3 Function 过程 .....           | 143 |
| 4.1.1 If 条件语句 .....                    | 83  | 6.3.1 建立 Function 过程 .....      | 143 |
| 4.1.2 Select ...Case 语句 .....          | 88  | 6.3.2 调用 Function 过程 .....      | 145 |
| 4.1.3 条件函数 .....                       | 91  | 6.4 不同模块间的过程调用 .....            | 146 |
| 4.2 循环结构 .....                         | 92  | 6.5 参数传送 .....                  | 146 |
| 4.2.1 For 循环控制结构 .....                 | 92  | 6.5.1 形参与实参 .....               | 147 |
| 4.2.2 当循环控制结构 .....                    | 97  | 6.5.2 按值传递参数 .....              | 147 |
| 4.2.3 Do 循环控制结构 .....                  | 100 | 6.5.3 按地址传递参数 .....             | 147 |
| 4.3 多重循环 .....                         | 104 | 6.5.4 数组参数的传送 .....             | 150 |
| 本章习题 .....                             | 106 | 6.6 递归过程 .....                  | 151 |
| <b>第 5 章 数 组</b> .....                 | 109 | 6.6.1 递归的概念 .....               | 151 |
| 5.1 数组的概念 .....                        | 109 | 6.6.2 递归子过程和递归函数<br>.....       | 152 |
| 5.1.1 数组的定义(声明) .....                  | 109 | 6.7 可选参数与可变参数 .....             | 153 |
| 5.1.2 数组函数 LBound()和 UBound()<br>..... | 111 | 6.7.1 可选参数 .....                | 153 |
| 5.1.3 变体型数组 .....                      | 112 | 6.7.2 可变参数 .....                | 155 |
| 5.2 静态数组与动态数组 .....                    | 113 | 6.8 对象参数 .....                  | 156 |
|                                        |     | 6.8.1 窗体参数 .....                | 156 |
|                                        |     | 6.8.2 控件参数 .....                | 157 |

|                                    |     |                                  |     |
|------------------------------------|-----|----------------------------------|-----|
| 6.9 键盘与鼠标事件过程 .....                | 160 | 8.1.2 目录列表框 .....                | 195 |
| 6.9.1 键盘事件过程 .....                 | 160 | 8.1.3 文件列表框 .....                | 196 |
| 6.9.2 鼠标事件过程 .....                 | 163 | 8.1.4 组合使用文件管理控件<br>.....        | 197 |
| 6.10 变量的作用域 .....                  | 170 | 8.1.5 执行文件 .....                 | 199 |
| 6.10.1 局部变量 .....                  | 170 | 8.2 通用对话框 .....                  | 200 |
| 6.10.2 模块级变量 .....                 | 171 | 8.2.1 概述 .....                   | 200 |
| 6.10.3 全局变量 .....                  | 171 | 8.2.2 文件对话框 .....                | 201 |
| 6.10.4 同名变量 .....                  | 171 | 8.2.3 其他对话框 .....                | 204 |
| 本章习题 .....                         | 172 | 8.2.4 通用对话框控件的应用<br>.....        | 206 |
| <b>第7章 数据文件</b> .....              | 173 | 本章习题 .....                       | 210 |
| 7.1 数据文件处理 .....                   | 173 | <b>第9章 图形处理</b> .....            | 212 |
| 7.1.1 数据文件概述 .....                 | 173 | 9.1 坐标系统 .....                   | 212 |
| 7.1.2 访问文件的语句和函数<br>.....          | 174 | 9.1.1 默认坐标系统 .....               | 212 |
| 7.2 顺序文件 .....                     | 179 | 9.1.2 自定义坐标系 .....               | 213 |
| 7.2.1 顺序文件的写操作 .....               | 179 | 9.2 色彩函数 .....                   | 214 |
| 7.2.2 顺序文件的读操作 .....               | 182 | 9.3 图形控件 .....                   | 215 |
| 7.3 随机文件 .....                     | 187 | 9.3.1 Shape(形状)控件 .....          | 215 |
| 7.3.1 变量声明 .....                   | 187 | 9.3.2 Line(直线)控件 .....           | 217 |
| 7.3.2 随机文件的打开 .....                | 188 | 9.3.3 PictureBox(图片框)控件<br>..... | 218 |
| 7.3.3 随机文件的写操作 .....               | 188 | 9.3.4 Image(图像框)控件 .....         | 221 |
| 7.3.4 随机文件的读操作 .....               | 188 | 9.4 绘图方法 .....                   | 222 |
| 7.3.5 增加、删除随机文件中的记录<br>.....       | 192 | 9.4.1 Pset(画点)方法 .....           | 222 |
| 7.4 二进制文件 .....                    | 193 | 9.4.2 Line(画线或矩形)方法 .....        | 222 |
| 本章习题 .....                         | 193 | 9.4.3 Circle(画圆)方法 .....         | 223 |
| <b>第8章 文件管理与通用对话框控<br/>件</b> ..... | 194 | 9.5 应用举例 .....                   | 224 |
| 8.1 文件管理控件 .....                   | 194 | 本章习题 .....                       | 229 |
| 8.1.1 驱动器列表框 .....                 | 194 | <b>参考文献</b> .....                | 231 |



【微信扫码】

# 第 1 章 Visual Basic 程序开发环境

## 1.1 可视化与事件驱动编程机制

Visual Basic(以下简称 VB)是用于开发和创建 Windows 操作平台下具有图形用户界面(Graphical User Interface, GUI)应用程序的强有力工具之一。作为一种可视化、基于事件驱动、面向对象(Object-oriented Programming, OOP)的编程工具,具有易学易用且功能强大的特点。

### 1.1.1 编程的可视化

VB 提供了可视化的编程工具,通过把 Windows 界面设计的复杂性“封装”起来,编程人员不必为界面设计而编写大量代码,只需要按设计要求的界面布局,采用系统提供的工具(控件),在屏幕上画出各种“控件”,并设置这些对象的属性,VB 自动产生界面设计代码,程序设计人员只需要实现程序功能的那部分代码。也就是说,编程工作是在图形用户界面上进行,在开发过程中实现了“所见即所得”。之所以叫作“可视”,用户只要看到 VB 的界面就会明白,无需编写大量代码去描述界面元素的外观和位置,把预先建立的可视对象拖到屏幕上即可。若要建立一个命令按钮,只要选择命令按钮的工具图标,用鼠标在所需的位置上拖拽放置就可以完成。开发者在开发的过程中,就能让用户看到当前开发的部分成果,容易提出修改意见。而这些成果想要在传统的面向过程的编程方法下实现,则要经过相当复杂的工作,通常要等程序全部完成并调试正确后才能演示。

### 1.1.2 事件驱动的编程机制

所谓事件,就是使各个对象进入活动状态(称为“激活”)的一种动作或操作。在事件驱动的应用程序中,程序代码根据要求分别组织到不同的事件过程中,代码不是按照预定的路径执行,而是在响应不同的事件时驱动不同的事件代码,以此来控制对象的行为。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,甚至由应用程序本身的消息触发。程序如何运行的控制权交给了用户,即使每次执行同一个程序的过程中,用户可能触发的事件次序也不一定完全一样。这就是事件驱动方式的应用程序设计原理。

尽管 VB 中的对象能够自动识别预定义的事件集,但必须通过代码判定它们是否响应具体事件以及如何响应具体事件,代码(即事件过程)与每个事件对应。为了让窗体或

控件响应某个事件,必须把代码放入这个对象的事件过程之中。

对象所能识别的事件类型有很多,多数类型为大多数对象所共有。例如,大多数对象都能识别 Click 事件,即单击事件;如果单击窗体,则执行窗体的单击事件过程中的代码;如果单击命令按钮,则执行命令按钮的单击事件过程中的代码。此外,某些事件可以在运行期间触发。例如,当在运行期间改变文本框中的文本时,将引发文本框的 Change 事件,如果 Change 事件过程中含有代码,则执行这些代码。

## 1.2 Visual Basic 的启动与退出

### 1.2.1 VB 6.0 的启动

通过“开始”菜单启动 VB 6.0,操作步骤为:

(1) 单击 Windows 桌面任务栏的“开始”按钮,弹出“开始”菜单,将鼠标指针指向“程序”选项,在“程序”项的级联菜单中选中“Microsoft Visual Basic 6.0 中文版”,然后在其打开的下级级联菜单中将光标条定位在“Microsoft Visual Basic 6.0 中文版”命令上。

(2) 单击鼠标左键,屏幕出现如图 1-1 所示的 VB 6.0 启动画面。

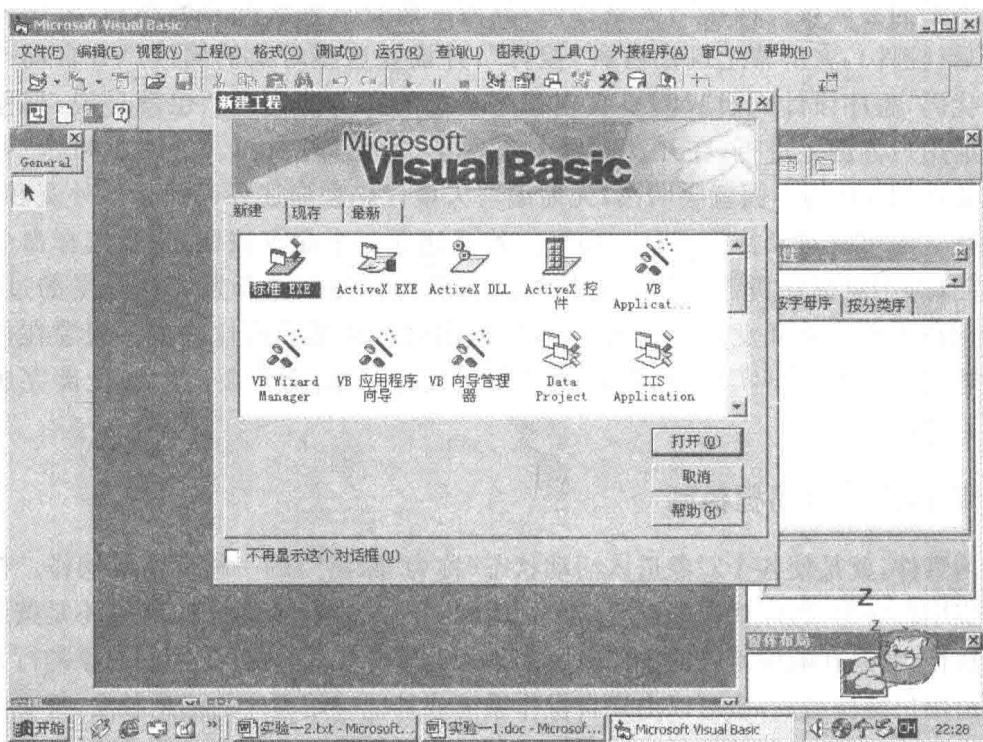


图 1-1 启动 VB 6.0

(3) 要建立一个新的工程,选择“新建”选项卡,从中选择“标准 EXE”项(默认),然后单击“打开”按钮,进入如图 1-2 所示的 VB 6.0 应用程序集成开发环境。

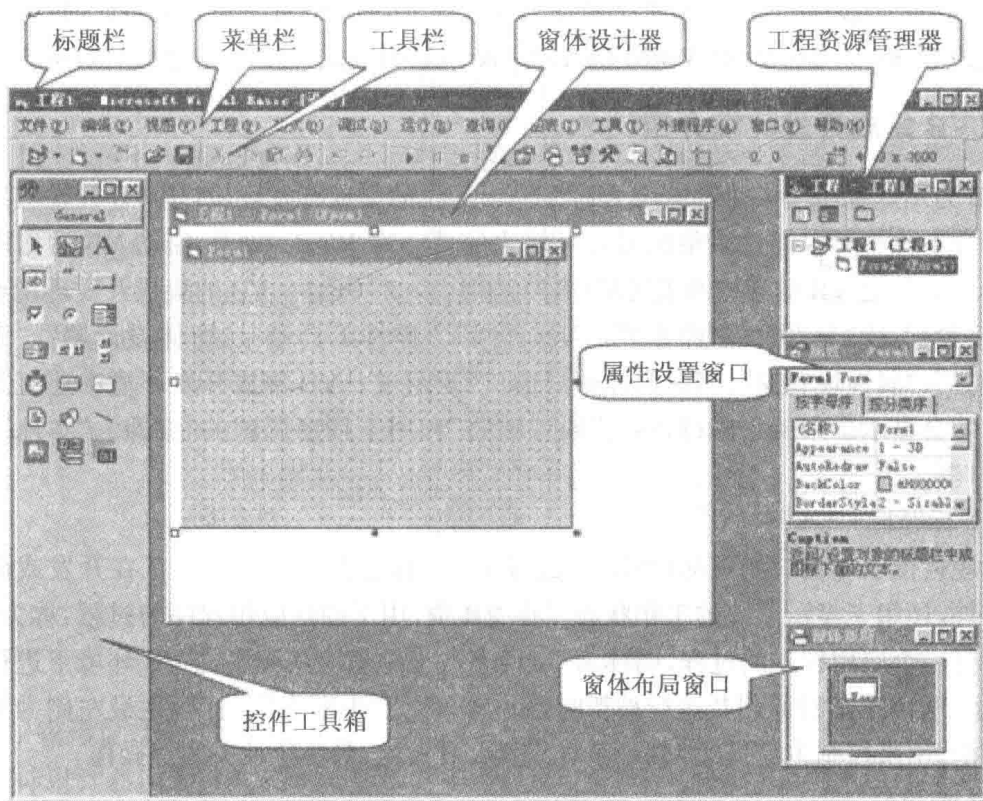


图 1-2 VB 6.0 应用程序集成开发环境

## 1.2.2 VB 6.0 的退出

在图 1-2 所示的 VB 6.0 应用程序集成开发环境窗口中,从“文件”菜单中选择“退出”命令,或双击窗口控制菜单图标,或单击窗口关闭按钮均可退出 VB 6.0。在退出时,系统可能会提示用户保存工程文件和窗体文件,有关保存文件的操作在后续内容进行详细说明。

## 1.3 Visual Basic 集成开发环境(IDE)的组成

### 1.3.1 VB 的三种工作模式

(1) 设计模式:启动 VB,选择新建一个新的工程,进入如图 1-2 所示的 VB 6.0 集成环境。此时,标题栏中的标题为“工程 1—Microsoft Visual Basic[设计]”,表明集成开发环境处于设计模式。

(2) 运行模式:用鼠标单击工具栏中的“!”按钮(启动),或在“运行”菜单中选择“启动”命令,此时,标题栏中的标题为“工程 1—Microsoft Visual Basic[运行]”,表明集成开发环境处于运行模式。

(3) 中断模式:在“运行”菜单中选择“中断”命令,或按(Ctrl+Break)键,此时,标题栏中的标题为“工程 1—Microsoft Visual Basic[break]”,表明集成开发环境处于中断模式。

### 1.3.2 VB 集成开发环境介绍

VB 为用户提供了一个功能强大而又易于操作的集成开发环境,用 VB 开发应用程序的大部分工作都可以通过该集成开发环境来完成。在 Windows 下,启动 VB 后出现在屏幕上的画面就是 VB 的集成开发环境(IDE)(如图 1-2 所示)。VB 的集成开发环境也称为 VB 的主窗口,由“标题栏”、“菜单栏”、“工具栏”、“控件工具箱”、“窗体设计器”、“工程资源管理器”、“属性设置窗口”和“窗体布局窗口”等组成。VB 集成开发环境中还有几个在必要时才会显示出来的子窗口,即“代码编辑器”和用于程序调试的“立即”、“本地”和“监视”窗口等。

#### (1) 标题栏

标题栏位于主窗口的顶部(如图 1-2 所示)。标题栏除了可显示正在开发或调试的工程名外,还用于显示系统的工作状态。在 VB 中,用于创建应用程序的过程,称为“设计态”;运行一个应用程序的过程,则称为“运行态”;当一个应用程序在 VB 环境下进行调试(即试运行),由于某种原因其运行被暂时终止时,称为“中断态”。标题栏最左侧为系统控制菜单框,用来控制主窗口的大小、移动、还原、最大化、最小化及关闭等操作。

#### (2) 菜单栏

菜单栏位于标题栏的下面(如图 1-2 所示)。VB 的菜单栏除了提供标准的“文件”、“编辑”、“视图”、“窗口”和“帮助”菜单之外,还提供了编程专用的功能菜单,如“工程”、“格式”、“调试”、“运行”、“查询”、“图表”、“工具”和“外接程序”等。

#### (3) 工具栏

工具栏一般位于菜单栏的下面(如图 1-2 所示)。VB 的工具栏包括有“标准”、“编辑”、“窗体编辑器”和“调试”四组工具栏。每个工具栏都由若干命令按钮组成,在编程环境下提供对于常用命令的快速访问。在没有进行相应设置的情况下,启动 VB 之后只显示“标准”工具栏。“编辑”、“窗体编辑器”和“调试”三个工具栏在需要使用的时候可通过选择“视图”菜单的“工具栏”命令中的相应工具栏名称来显示,也可通过鼠标右击“标准”工具栏的空白部分,从打开的弹出式菜单中选择需要的工具栏名称来显示。

#### (4) 控件工具箱

控件工具箱又称工具箱,一般位于 VB 主窗口的左边(如图 1-2 所示)。它提供的是软件开发人员在设计应用程序界面时需要使用的常用工具(控件)。这些控件以图标的形式存放在工具箱中,软件开发人员在设计应用程序时,使用这些控件在窗体上“画”出应用程序的界面。常用控件的图标和名称如图 1-3 所示。

工具箱除了最常用的控件以外,根据设计程序界面的需要也可以向工具箱中添加新的控件,添加新控件可以通过选择“工程”菜单中的“部件”命令或通过在工作箱中右击鼠标,在弹出菜单中选择“部件”命令来完成。

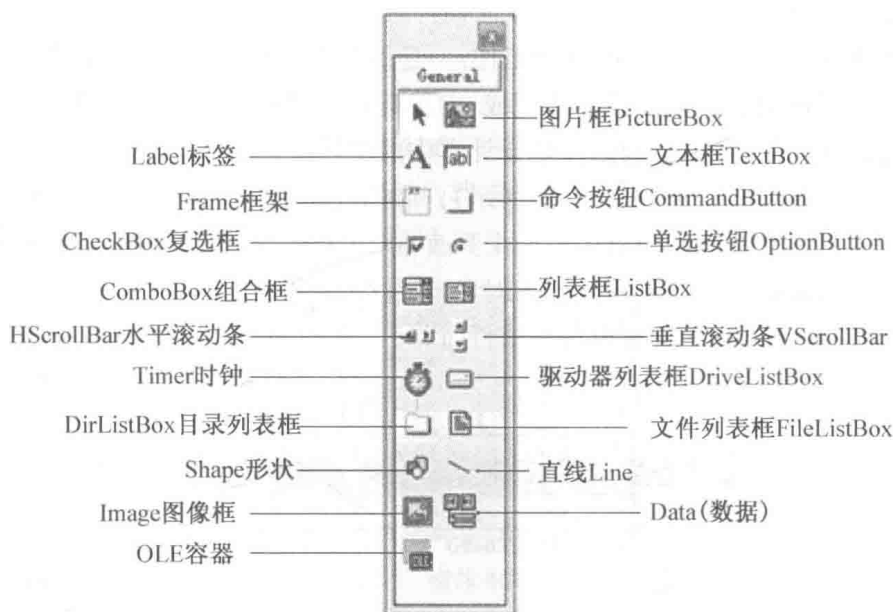


图 1-3 VB 的控件工具箱

#### (5) 窗体设计器

窗体设计器位于 VB 主窗口的中间(如图 1-2 所示)。它是一个用于设计应用程序界面的自定义窗口。应用程序中每一个窗体都有自己的窗体设计器。当启动 VB 开始创建一个新工程时,窗体设计器和它中间的初始窗体“Form1”一道出现。要在应用程序中添加其他窗体,可单击工具栏上的“添加窗体”按钮。

#### (6) 属性设置窗口

属性设置窗口一般位于窗体设计器的右方(如图 1-2 和图 1-4 所示)。它主要用来在设计界面时,为所选中的窗体和窗体上的各个对象设置初始属性值。它由标题栏、“对象”列表框、“属性”列表框及属性说明几部分组成。属性设置窗口的标题栏中标有窗体的名称。用鼠标单击标题栏下的“对象”列表框右侧的按钮,打开其下拉式列表框,可从中选取本窗体内的各个对象,对象选定后,下面的属性列表框中就列出与该对象有关的各个属性及其设定值。

属性窗口设有“按字母序”和“按分类序”两个选项卡,可分别将属性按字母或按分类顺序排列。当选中某一属性时,在下面的说明框里就会给出该属性的相关说明。



图 1-4 属性设置窗口

### (7) 编辑器

用 VB 开发应用程序,包括两部分工作:一是设计图形用户界面;二是编写程序代码。设计图形用户界面通过窗体设计器来完成;而代码编辑器的作用就是用来编写应用程序代码。设计程序时,当用鼠标双击窗体设计器中的窗体或窗体上的某个对象时,代码编辑器将显示在 VB 集成环境中(如图 1-5 所示)。应用程序的每个窗体和标准模块都有一个单独的代码编辑器。代码编辑器中有两个列表框,一个是“对象”列表框,另一个是“事件”列表框。从列表框中选定要编写代码的对象(若是公共代码段,则选“通用”),再选定相应的事件,则可以非常方便地为对象编写事件过程。

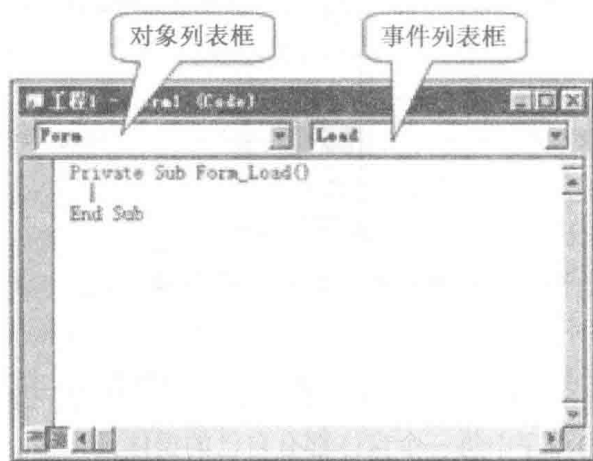


图 1-5 代码编辑器

### (8) 工程资源管理器

工程资源管理器又称为工程浏览器,一般位于窗体设计器的右上方(如图 1-2 和图 1-6 所示)。它列出了当前应用程序中包含的所有文件清单。一个 VB 应用程序也称为一个工程,由一个工程文件(.vbp)和若干个窗体文件(.frm)、标准模块文件(.bas)与类模块文件(.cls)等其他类型文件组成。工程资源管理器窗口上有一个小工具栏,上面的三个按钮分别用于查看代码、查看对象和切换文件夹。在工程资源管理器窗口中选定对象,单击

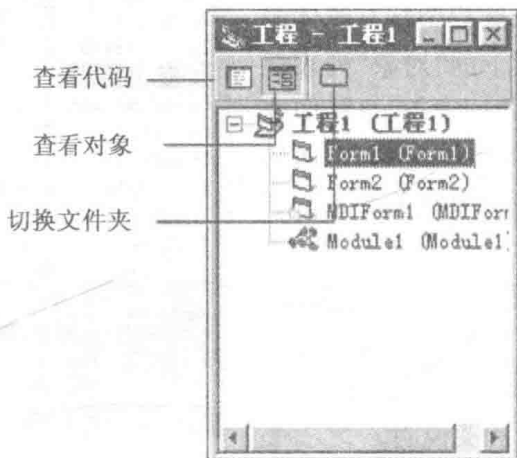


图 1-6 工程资源管理器

“查看对象”按钮,即可在窗体设计器中显示所要查看的窗体对象;单击“查看代码”按钮,则会出现该对象的“代码编辑器”窗口。

#### (9) 窗体布局窗口

窗体布局窗口位于窗体设计器的右下方(如图 1-2 和图 1-7 所示)。在设计时通过鼠标右击表示屏幕的小图像中的窗体图标,将会弹出一个菜单,选择菜单中的相关命令项,可设置程序运行时窗体在屏幕上的位置。

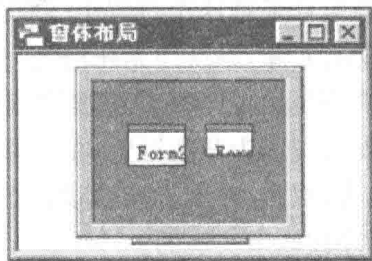


图 1-7 窗体布局窗口

## 1.4 对象及其操作

用 Visual Basic 进行应用程序设计,实际上是与一组标准对象进行交互的过程,就是把这些标准对象按照要求放置在合适的位置,并设计程序完成相应的功能。因此,准确理解和认识对象的概念,是设计 Visual Basic 应用程序的重要一步。

### 1.4.1 对象(Object)

对象是 Visual Basic 中的重要概念,离开了对象,Visual Basic 的程序设计将无从谈起。在面向对象程序设计中,“对象”是系统中的基本运行实体。在 Visual Basic 中,对象分为两类:一类是由系统设计好的,称为预定义对象,可以直接使用或对其操作;另一类由用户定义,可以像 C++ 一样建立用户自己的对象。本书主要讨论两种最基本的预定义对象:窗体(Form)和控件(Control)。

Form: 窗体或称表单,其实指的就是窗口。

Control: 控件,指的是各种按钮、标签等。

对象类是指具有某一类特殊性质(属性)和行为方式(方法)的实体。建立一个对象以后,其操作通过与该对象有关的属性、事件和方法描述。

### 1.4.2 属性(Property)

属性是用来描述对象的特性,比如姓名、性别、民族、籍贯都是你这个对象的属性。VB 中的每个对象都有它的属性,并且“Name”属性是共有的,有了“Name”属性才可以在程序中进行调用。对象的属性是可以改变的,可以在窗体界面设计时完成,也可以在程序运行中改变,这取决于用户的需要,但有些属性是只读的,它只能在设计态的时候改变。

我们将在以后的实例中具体介绍各个对象属性的作用。

### 1.4.3 事件(Event)

事件是发生在对象上的动作。比如敲桌子是一个事件,它是发生在桌子这个对象上的一个动作。比如 Click、Dblick 或 GotFocus 是发生在文本框控件上的事件。然而事件的发生不是随意的,某些事件仅发生在某些对象上而已,比如“逃避早操被抓住”可以发生在学生这个对象上,但它不会发生在老师这个对象上。我们可以把一个应用系统看成是由若干个对象组成的。但是,单有对象,系统还是死板的(或静态的),无法做出各种动作,无法运行。就像一个话剧,单有演员出现还不够,还需要设计每个演员在规定条件下应有的台词和动作。因此,软件开发应该为某些对象,在某种事件的触发下,设计程序代码,完成指定的动作。通常鼠标单击、双击、右击、拖动等都可以作为事件,指定对象响应指定事件所做的动作就是靠编写程序代码来实现的。

### 1.4.4 方法(Method)

“方法”是对象可以进行的动作或行为。人们可以通过“方法”使对象改变行为或做某种动作。

这是一个直译,是一个较难理解的概念,它是对象本身内含的函数或过程,它也是一个动作,但不称作事件,在 VB 里,方法和事件的格式是这样的:

事件:

```
Private Sub 对象名_事件名
```

(事件内容)

```
End Sub
```

方法:

```
对象名.方法名
```

对于窗体,有隐藏、显示方法;文本框有刷新、设置焦点等方法。

所以方法是一个简单的不必知道细节的无法改变的事件,同样,方法也不是随意的,一些对象有一些特定的方法。如果以上概念你记不住,不要紧,实践中你会明白一切,请继续学习。

**注意:**许多事件伴随其他事件发生。例如,在双击事件发生时,MouseDown、MouseUp 和单击事件也会发生。

### 1.4.5 属性、方法和事件之间的关系

VB 对象具有属性、方法和事件。属性是描述对象的数据;方法告诉对象应做的事情;事件是对象所产生的事情,事件发生时编写代码进行处理。

VB 的窗体和控件是具有自己的属性、方法和事件的对象。可以把属性看作一个对象的性质,把方法看作对象的动作,把事件看作对象的响应。

日常生活中的对象,如小孩玩的气球同样具有属性、方法和事件。气球的属性包括可以看到的一些性质,如它的直径和颜色。其他一些属性描述气球的状态(充气的或未充气

的)或不可见的性质,如它的寿命。通过定义,所有气球都具有这些属性;这些属性也会因气球的不同而不同。

气球还具有本身所固有的方法和动作。如:充气方法(用氦气充满气球的动作),放气方法(排出气球中的气体)和上升方法(放手让气球飞走)。所有的气球都具备这些能力。

气球还有预定义的对某些外部事件的响应。例如,气球对刺破它的事件响应是放气,对放手事件的响应是升空。

在 VB 程序设计中,基本的设计机制就是:改变对象的属性、使用对象的方法、为对象事件编写事件过程。程序设计时要做的工作就是决定应更改哪些属性、调用哪些方法、对哪些事件做出响应,从而得到希望的外观和行为。

## 1.5 程序设计的一般步骤

一个 VB 程序也称为一个工程,由窗体、标准模块、自定义控件及应用所需的环境设置组成。开发步骤一般如下:

- ① 创建程序的用户界面
- ② 设置界面上各个对象的属性
- ③ 编写对象响应事件的程序代码
- ④ 保存工程
- ⑤ 测试应用程序,排除错误
- ⑥ 创建可执行程序

下面我们通过一个简单的例子来说明如何在 VB 环境下设计应用程序。这个例子很简单,但它展示了应用程序设计的全过程。

程序要求:图 1-8 是本例的程序界面。在窗口中的标签上有一行文字:“你好!”和一个命令按钮。用鼠标单击命令按钮,窗口中的文字就会自动变成图 1-9“欢迎学习 VB!”。

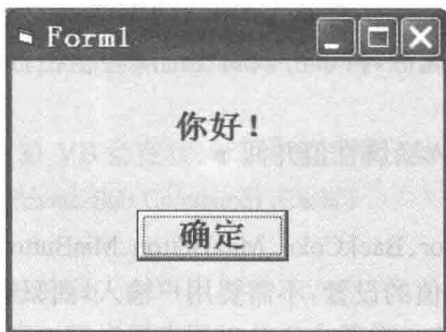


图 1-8 初始界面

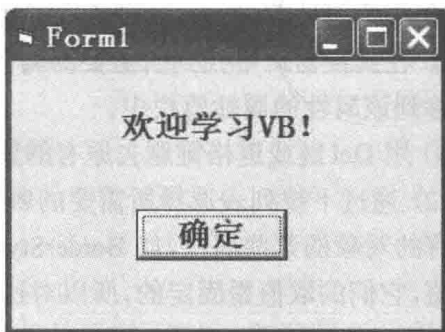


图 1-9 运行界面

### 1.5.1 创建用户界面

即设计窗体以及在窗体中放置控件和对象。主要完成以下工作:

- (1) 启动 VB,新建一个工程

可采用以下两种方法:

方法 1:从窗口的开始菜单中启动 VB 6.0,出现“新建工程”对话框,在该对话框中,选择“标准 EXE”,然后单击“打开”按钮。

方法 2:在 VB 集成开发环境中,在“文件”菜单中单击“新建工程”子菜单,然后在“新建工程”对话框中,选择“标准 EXE”,然后单击“打开”按钮。

此时,VB 给定一个默认的工程名称,叫作“工程 1”,而这个工程一开始只含有一个默认的窗体叫“Form1”。

(2) 创建用户界面,向窗体中添加控件

本例需要向窗体上添加一个“Label1”标签控件和一个“Command1”命令按钮控件。

第一步,用鼠标单击工具箱中的“标签控件”,然后将鼠标移动到“窗体设计器”中新建的窗体上,此时鼠标为十字状。

第二步,按下鼠标左键,向右下拖动鼠标,当大小适当时松开鼠标,此时就在该窗体上画出了一个“标签控件”。

然后重复以上第一步、第二步,向窗体中添加所需要的“命令按钮”。

## 1.5.2 设置对象属性

对于本例而言,需要将“Label1”标签和“Command1”命令按钮的“Caption”属性分别设置为“你好”和“确定”。可以通过以下方法和步骤完成。

对窗体和控件等对象进行属性的设置,可以在程序设计阶段进行,也可以通过程序代码在应用程序运行时修改它们的属性。

1. 在设计阶段利用属性窗口设置对象属性

在程序设计阶段,可以利用属性窗口设置对象的属性,由于不同的属性,VB 可能提供了不同的属性值的设置方法,所以,下面分三种情况进行介绍:

(1) 在属性窗口中直接键入新属性值

① 在窗体设计器中选择某一控件。

② 激活属性窗口。

③ 在属性窗口中找到所需要的属性,单击该属性,再单击该属性的属性值栏,即把插入点移到该属性的属性值栏中。

④ 用 Del 键或退格键删去原有的属性值,输入新属性值并回车。

(2) 通过下拉列表选择所需要的属性值

有的对象的某些属性,如 BorderStyle、ForeColor、BackColor、MaxButton、MinButton 等的属性值,它们的取值是固定的,所以对这样的属性值的设置,不需要用户输入,而只需从属性窗口选择即可,其方法是:

① 在窗体设计器中选择某一控件。

② 激活属性窗口。

③ 在属性窗口中找到所需要的属性,单击该属性,可见该属性的属性值的右端出现一个向下的箭头(即:下拉列表)。

④ 单击该下拉列表的右端箭头,可见列表中将显示出该属性所有可能的取值。