



# Visual Basic

中文版

## 程序设计教程

唐 兵 乔丰立 李桂花 等编著



零距离电脑培训学校

编程系列

# Visual Basic 中文版程序设计教程

唐 兵 乔丰立 李桂花 等编著



机械工业出版社

本书系统、全面地讲述了 Visual Basic 6.0 中文版的使用方法,内容包括:认识 Visual Basic、Visual Basic 语言基础、窗体的设计和使用、使用 Visual Basic 标准控件、创建菜单和工具栏、对话框的设计与使用、文件管理与操作、使用 Windows 通用控件、数据库编程、多媒体编程、代码调试和错误处理。附录中还介绍了 Visual Basic 6.0 中文版的安装方法和 Visual Basic 的常用内部函数。

本书结构新颖,内容全面,循序渐进,范例丰富,每个单元都有大量的测试题,可以作为 Visual Basic 6.0 应用程序开发的培训教材,也可以作为广大 Visual Basic 爱好者的参考书。

### 图书在版编目(CIP)数据

Visual Basic 中文版程序设计教程/唐兵等编著. —北京:  
机械工业出版社, 2005.1

(零距离电脑培训学校编程系列)

ISBN 7 - 111 - 15349 - 9

I . V... II . 唐... III . BASIC 语言 - 程序设计 - 教材 IV . TP312

中国版本图书馆 CIP 数据核字(2004)第 099822 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)

策 划: 胡毓坚

责任编辑: 时 静

责任印制: 施 红

北京铭成印刷有限公司印刷·新华书店北京发行所发行

2005 年 1 月第 1 版·第 1 次印刷

787mm × 1092mm  $\frac{1}{16}$  · 18 印张 · 441 千字

0001—5000 册

定价: 25.00 元

凡购本图书,如有缺页、倒页、脱页,由本社发行部调换  
本社购书热线电话 (010) 68993821、88379646

封面无防伪标均为盗版

# 出版说明

近几年来，电脑在我国迅速普及，人们的日常生活、娱乐和工作越来越离不开电脑。能够熟练使用电脑也是许多行业对从业者的基本要求。

目前，我国有大量电脑初级用户，他们迫切要求掌握电脑操作的基本方法；还有许多已有一定电脑基础知识的中级用户，希望学会使用与自身工作密切相关的软件。但是在紧张的工作之后，多数人不可能花费太多的时间来系统地学习电脑知识。基于以上背景，我社邀请国内著名计算机职业教育学校的资深老师，为电脑初、中级用户编写了这套“零距离电脑培训学校”丛书。

本套丛书紧紧围绕“短期培训”这个中心，尽量将基础知识与基本技能贯穿于基本操作和应用能力教学之中，书中列举了大量实例，鼓励读者在练中学。丛书强调“不求全、不求精、只求会”，对每一种软件或技术不要求学全，只要学会其中最重要的、与学习者的工作或专业联系最密切的内容就可以。这套丛书通俗易懂、图文并茂，便于读者迅速掌握所学知识。

为了便于读者自学以及培训班授课，我们为每本书配了电子教案，读者可以在我社网站（<http://www.cmpbook.com>）免费下载。

本套丛书覆盖了电脑应用的大部分领域。今后我们会不断补充新的图书，以满足广大读者的需求。

机械工业出版社

# 前 言

由于 Visual Basic 功能强大、简单易用而拥有广泛的用户，尤其是在数据库、多媒体、商用系统的用户界面等方面，Visual Basic 更是具有其他开发环境不可能取代的优势。升级到 6.0 版后，其功能得到了极大的增强：汉化后的 Visual Basic 6.0 中文版，由于有全新的中文界面，更方便了用户的使用。

Visual Basic 6.0 有三种版本，各自满足不同的开发需要：“Visual Basic 学习版”让编程人员很容易地创建功能强大的 Microsoft Windows 和 Windows NT(R) 应用程序；“Visual Basic 专业版”向计算机专业人员提供了一套功能完整的工具，便于他们开发解决方案，专业版包含了学习版的所有功能。“Visual Basic 企业版”允许专业人员以小组的形式来创建强健的分布式应用程序，它包括专业版的所有特性。

本书以 Visual Basic 6.0 三种版本的公共方面为主题，系统、全面地讲述了 Visual Basic 6.0 中文版的使用方法。全书内容包括：认识 Visual Basic、Visual Basic 语言基础、窗体的设计和使用、使用 Visual Basic 标准控件、创建菜单和工具栏、对话框的设计与使用、文件管理与操作、使用 Windows 通用控件、数据库编程、多媒体编程、代码调试和错误处理。附录中还介绍了 Visual Basic 6.0 中文版的安装方法和 Visual Basic 的常用内部函数。

本书是一本容易理解、容易使用的 Visual Basic 6.0 中文版培训教材，几乎包括了 Visual Basic 6.0 的所有主题内容。本书假定读者以前从未使用过 Visual Basic 编写程序。对于使用过 Visual Basic 的早期版本或其他结构化编程语言的读者来说，本书内容更容易理解。如果是从头开始学习，也能很快掌握 Visual Basic 6.0 中文版，并能充分利用其强大的数据访问特性、ActiveX (TM) 技术、Internet 能力等开发出精彩、完美的 Windows/Windows NT 应用程序。

本书第 1、2、3、4 课由唐兵编写，第 5、6 课由乔丰立编写，第 7、8 课由李桂花编写，第 9、10、11 课由郝燕文编写，全书由卢堃统编。另外，张江涛、刘伟、尹建民、张海霞、吴建文、智雨青、刘旭、赵磊、徐日强、范翠丽、薛年喜、王建平、李士良等也参加了部分内容的编写及素材整理工作，在此一并表示感谢。

由于 Visual Basic 涉及的内容很多，加之作者水平有限，书中不足之处敬请广大读者批评指正。读者可将意见通过网站 [www.bykw.com](http://www.bykw.com) 反馈给我们。

编 者

# 目 录

出版说明

前言

|                                     |    |
|-------------------------------------|----|
| 第 1 课 认识 Visual Basic .....         | 1  |
| 1.1 课前导读 .....                      | 1  |
| 1.2 课堂教学 .....                      | 1  |
| 1.2.1 Visual Basic 简介 .....         | 1  |
| 1.2.2 Visual Basic 的集成开发环境 .....    | 2  |
| 1.2.3 菜单和工具栏 .....                  | 3  |
| 1.2.4 工具箱 .....                     | 3  |
| 1.2.5 设计窗体 .....                    | 4  |
| 1.2.6 代码窗口 .....                    | 4  |
| 1.2.7 工程窗口 .....                    | 5  |
| 1.2.8 属性窗口 .....                    | 5  |
| 1.2.9 窗体布局窗口 .....                  | 5  |
| 1.2.10 Visual Basic 的对象 .....       | 5  |
| 1.2.11 对象的属性、事件和方法 .....            | 6  |
| 1.2.12 Visual Basic 的控件 .....       | 8  |
| 1.3 上机操作 .....                      | 10 |
| 1.3.1 创建第一个 Visual Basic 应用程序 ..... | 10 |
| 1.3.2 定制 Visual Basic 的开发环境 .....   | 15 |
| 1.4 入门常见问题解答 .....                  | 16 |
| 1.5 课后作业 .....                      | 16 |
| 第 2 课 Visual Basic 语言基础 .....       | 18 |
| 2.1 课前导读 .....                      | 18 |
| 2.2 课堂教学 .....                      | 18 |
| 2.2.1 代码模块 .....                    | 18 |
| 2.2.2 注释 .....                      | 19 |
| 2.2.3 分行和续行 .....                   | 20 |
| 2.2.4 数据类型 .....                    | 20 |
| 2.2.5 变量与常数 .....                   | 24 |
| 2.2.6 运算符与表达式 .....                 | 30 |
| 2.2.7 数组 .....                      | 34 |
| 2.2.8 控制结构 .....                    | 38 |
| 2.2.9 子过程和函数 .....                  | 45 |

|                                       |           |
|---------------------------------------|-----------|
| 2.2.10 培养良好的编程习惯 .....                | 52        |
| 2.3 上机操作 .....                        | 56        |
| 2.3.1 使用循环 .....                      | 56        |
| 2.3.2 使用数组 .....                      | 57        |
| 2.3.3 使用子过程和函数 .....                  | 58        |
| 2.4 Visual Basic 语言常见问题解答 .....       | 60        |
| 2.5 课后作业 .....                        | 61        |
| <b>第3课 窗体的设计和使用</b> .....             | <b>64</b> |
| 3.1 课前导读 .....                        | 64        |
| 3.2 课堂教学 .....                        | 64        |
| 3.2.1 创建窗体 .....                      | 64        |
| 3.2.2 设置窗体属性 .....                    | 65        |
| 3.2.3 窗体的事件 .....                     | 68        |
| 3.2.4 窗体的方法 .....                     | 71        |
| 3.2.5 窗体的生命周期 .....                   | 72        |
| 3.2.6 在窗体上添加控件 .....                  | 76        |
| 3.2.7 在窗体上绘图 .....                    | 77        |
| 3.2.8 界面设计的基本原则 .....                 | 82        |
| 3.3 上机操作 .....                        | 85        |
| 3.3.1 创建一个多窗体程序 .....                 | 85        |
| 3.3.2 在窗体上绘图 .....                    | 86        |
| 3.4 窗体设计和使用中的常见问题解答 .....             | 88        |
| 3.5 课后作业 .....                        | 89        |
| <b>第4课 使用 Visual Basic 标准控件</b> ..... | <b>91</b> |
| 4.1 课前导读 .....                        | 91        |
| 4.2 课堂教学 .....                        | 91        |
| 4.2.1 命令按钮控件 .....                    | 91        |
| 4.2.2 标签控件 .....                      | 93        |
| 4.2.3 文本框控件 .....                     | 96        |
| 4.2.4 图像框控件 .....                     | 100       |
| 4.2.5 滚动条控件 .....                     | 104       |
| 4.2.6 定时器控件 .....                     | 105       |
| 4.2.7 其他常用控件 .....                    | 106       |
| 4.3 上机操作 .....                        | 112       |
| 4.3.1 自制简易计算器 .....                   | 112       |
| 4.3.2 利用滚动条控件改变窗体颜色 .....             | 115       |
| 4.3.3 利用定时器控件实现简单动画 .....             | 116       |
| 4.4 Visual Basic 标准控件常见问题解答 .....     | 117       |
| 4.5 课后作业 .....                        | 118       |

|                                            |     |
|--------------------------------------------|-----|
| <b>第 5 课 创建菜单和工具栏</b>                      | 120 |
| 5.1 课前导读                                   | 120 |
| 5.2 课堂教学                                   | 120 |
| 5.2.1 创建下拉菜单                               | 120 |
| 5.2.2 创建弹出式菜单                              | 123 |
| 5.2.3 为菜单添加功能代码                            | 125 |
| 5.2.4 使用 <code>Imagelist</code> 控件         | 125 |
| 5.2.5 创建工具栏                                | 128 |
| 5.2.6 为工具栏添加功能代码                           | 133 |
| 5.3 上机操作                                   | 133 |
| 5.3.1 创建和使用菜单                              | 133 |
| 5.3.2 创建和使用工具栏                             | 141 |
| 5.4 菜单和工具栏常见问题解答                           | 144 |
| 5.5 课后作业                                   | 145 |
| <b>第 6 课 对话框的设计与使用</b>                     | 146 |
| 6.1 课前导读                                   | 146 |
| 6.2 课堂教学                                   | 146 |
| 6.2.1 预定义对话框                               | 146 |
| 6.2.2 打开文件对话框                              | 150 |
| 6.2.3 保存文件对话框                              | 152 |
| 6.2.4 字体对话框                                | 154 |
| 6.2.5 颜色对话框                                | 155 |
| 6.2.6 打印对话框                                | 156 |
| 6.2.7 自定义对话框                               | 157 |
| 6.3 上机操作                                   | 161 |
| 6.3.1 使用字体对话框                              | 161 |
| 6.3.2 使用颜色对话框                              | 162 |
| 6.4 对话框使用常见问题解答                            | 163 |
| 6.5 课后作业                                   | 164 |
| <b>第 7 课 文件管理与操作</b>                       | 165 |
| 7.1 课前导读                                   | 165 |
| 7.2 课堂教学                                   | 165 |
| 7.2.1 <code>File System Object</code> 模型介绍 | 165 |
| 7.2.2 <code>FSO</code> 对象模型编程              | 166 |
| 7.2.3 使用 <code>FSO</code> 处理驱动器和文件夹        | 168 |
| 7.2.4 使用 <code>FSO</code> 处理文件             | 170 |
| 7.2.5 传统的文件 <code>I/O</code> 语句和函数         | 172 |
| 7.2.6 使用顺序文件访问                             | 174 |
| 7.2.7 使用随机文件访问                             | 176 |

|                                    |     |
|------------------------------------|-----|
| 7.2.8 使用二进制文件访问 .....              | 178 |
| 7.2.9 文件系统控件 .....                 | 179 |
| 7.3 上机操作 .....                     | 182 |
| 7.3.1 使用数据文件 .....                 | 182 |
| 7.3.2 创建文件管理程序 .....               | 185 |
| 7.4 文件操作的常见问题解答 .....              | 189 |
| 7.5 课后作业 .....                     | 190 |
| <b>第 8 课 使用 Windows 通用控件</b> ..... | 191 |
| 8.1 课前导读 .....                     | 191 |
| 8.2 课堂教学 .....                     | 191 |
| 8.2.1 选项卡 (TabStrip) 控件 .....      | 191 |
| 8.2.2 状态条 (StatusBar) 控件 .....     | 196 |
| 8.2.3 进度条 (ProgressBar) 控件 .....   | 199 |
| 8.2.4 图形组合框控件 .....                | 201 |
| 8.3 上机操作: 使用选项卡控件 .....            | 204 |
| 8.4 Windows 通用控件使用常见问题解答 .....     | 205 |
| 8.5 课后作业 .....                     | 206 |
| <b>第 9 课 数据库编程</b> .....           | 207 |
| 9.1 课前导读 .....                     | 207 |
| 9.2 课堂教学 .....                     | 207 |
| 9.2.1 数据库的基本概念 .....               | 207 |
| 9.2.2 使用可视化数据管理器 .....             | 210 |
| 9.2.3 ADO 编程模型简介 .....             | 215 |
| 9.2.4 使用 ADO 数据控件 .....            | 221 |
| 9.2.5 使用数据绑定控件 .....               | 223 |
| 9.2.6 创建数据报表 .....                 | 226 |
| 9.3 上机操作: 创建数据库查询程序 .....          | 230 |
| 9.4 数据库常见问题解答 .....                | 231 |
| 9.5 课后作业 .....                     | 232 |
| <b>第 10 课 多媒体编程</b> .....          | 234 |
| 10.1 课前导读 .....                    | 234 |
| 10.2 课堂教学 .....                    | 234 |
| 10.2.1 多媒体的基本概念 .....              | 234 |
| 10.2.2 使用 OLE 开发多媒体程序 .....        | 236 |
| 10.2.3 使用 Animation 控件 .....       | 237 |
| 10.2.4 使用 Multimedia 控件 .....      | 240 |
| 10.3 上机操作: 使用 Multimedia 控件 .....  | 243 |
| 10.4 多媒体编程常见问题解答 .....             | 247 |
| 10.5 课后作业 .....                    | 247 |

|                                        |     |
|----------------------------------------|-----|
| <b>第 11 课 代码调试和错误处理</b> .....          | 249 |
| 11.1 课前导读 .....                        | 249 |
| 11.2 课堂教学 .....                        | 249 |
| 11.2.1 调试方法概述 .....                    | 249 |
| 11.2.2 错误处理概述 .....                    | 254 |
| 11.2.3 设计错误处理程序 .....                  | 255 |
| 11.2.4 测试错误处理 .....                    | 259 |
| 11.2.5 关闭错误处理 .....                    | 260 |
| 11.2.6 使用调试窗口 .....                    | 261 |
| 11.3 上机操作 .....                        | 262 |
| 11.3.1 监视程序变量 .....                    | 262 |
| 11.3.2 捕获和处理错误 .....                   | 263 |
| 11.4 程序调试常见问题解答 .....                  | 267 |
| 11.5 课后作业 .....                        | 267 |
| <b>附录</b> .....                        | 268 |
| <b>附录 A Visual Basic 6.0 的安装</b> ..... | 268 |
| A.1 Visual Basic 6.0 的安装系统要求 .....     | 268 |
| A.2 Visual Basic 6.0 的安装步骤 .....       | 268 |
| A.3 添加或删除 Visual Basic 6.0 部件 .....    | 269 |
| <b>附录 B Visual Basic 常用内部函数</b> .....  | 270 |

# 第 1 课 | 认识 Visual Basic

## 学习目标:

- Visual Basic 简介
- Visual Basic 的集成开发环境
- Visual Basic 的对象
- 对象的属性、方法和事件
- Visual Basic 的控件

## 学习重点:

- Visual Basic 的对象
- 对象的属性、方法和事件
- Visual Basic 的控件

## 1.1 课前导读

BASIC 编程语言产生于 1963 年,它是第一个让程序员把主要的编程工作集中在解决编程任务的方法和算法上,而不是集中在计算机硬件所需的建立和调试程序的方法和算法上的一门高级编程语言。Visual Basic 是在 BASIC 的基础上发展起来,虽然其发展已经远远超出了最初的设计,然而其基本原则仍然是一致的。众所周知,使用 C 或 C++进行 Microsoft Windows 编程是十分复杂的,而使用 Visual Basic 能够在很短的时间内就可实现同样的功能。Visual Basic 是极其高效的 Windows 软件开发工具,它可以经济快速地开发出程序员想得到的结果。

## 1.2 课堂教学

### 1.2.1 Visual Basic 简介

Microsoft Visual Basic 提供了开发 Microsoft Windows 应用程序的最迅速、最简捷的方法。不论是 Microsoft Windows 应用程序的资深专业开发人员还是初学者, Visual Basic 都能为他们提供整套工具,以方便开发应用程序。

何为 Visual Basic? “Visual”指的是可视化的图形用户界面 (GUI) 开发方法。不需



编写大量代码去描述界面元素的外观和位置，只要把预先建立的对象添加到屏幕上的某点即可。如果已使用过诸如 Paint 之类的绘图程序，则实际上已掌握了创建用户界面的必要技巧。

“Basic”指的是 BASIC (Beginners All-Purpose Symbolic Instruction Code) 语言，一种在计算技术发展历史上应用得最为广泛的语言之一。Visual Basic 在原有 BASIC 语言的基础上进一步发展，至今包含了数百条语句、函数及关键词，其中很多和 Windows GUI 有直接关系。专业人员可以用 Visual Basic 实现其他任何 Windows 编程语言的功能，而初学者只要掌握几个关键词就可以建立实用的应用程序。

Visual Basic 不仅是可视化 Basic 编程语言，而且是面向对象的编程语言。Visual Basic 编程系统和 VBA 都使用这一语言。Visual Basic Scripting Edition (VBScript) 是广泛使用的脚本语言，它是 Visual Basic 语言的子集。这样，在学习 Visual Basic 中得到的经验可应用到所有这些领域中。

从开发个人或小组使用的小工具，到大型企业应用系统，甚至通过 Internet 遍及全球的分布式应用程序，都可在 Visual Basic 提供的工具中找到需要的。

与 BASIC 相比，Visual Basic 增加了如下新的特性：

- 1) 功能强大，而且易学易用的集成开发环境，为开发功能强大的应用程序节省了许多时间。
- 2) 事件驱动模型和交互式开发方法，改变了代码的执行顺序和开发的传统步骤。
- 3) 面向对象的程序设计方法，简单的结构化程序设计语言，从而使 Visual Basic 设计语言成为初学者首选的程序设计语言。
- 4) 数据访问特性允许对包括 Microsoft SQL Server 和其他企业数据库在内的大部分数据库格式建立数据库和前端应用程序，以及可调整的服务器端部件。
- 5) 有了 ActiveX (TM) 技术就可使用其他应用程序提供的功能，例如 Microsoft Word 字处理器，Microsoft Excel 电子数据表及其他 Windows 应用程序。甚至可直接使用 VBP 或 VBE 创建的应用程序和对象。
- 6) Internet 能力强大，使得用户很容易在应用程序内通过 Internet 或 Intranet 访问文档和应用程序，或者创建 Internet 服务器应用程序。
- 7) 已完成的应用程序是使用 Visual Basic 虚拟机真正的 exe 文件，用户可以自由发布。

## 1.2.2 Visual Basic 的集成开发环境

Visual Basic 的工作环境常常是指集成开发环境 (IDE)，这是因为它在一个公共环境里集成了许多不同的功能，例如，设计、编辑、编译和调试。在大多传统开发工具中，每个功能都是以一个独立的程序运行，并都有自己的界面。

当运行 Visual Basic 安装程序时，允许将程序项置于已存在的程序组中，或在 Windows 中为 Visual Basic 创建一个新的程序组和程序项。这时可从 Windows 启动 Visual Basic。

要从 Windows 启动 Visual Basic，请按照以下步骤执行：

- 1) 单击任务条上的【开始】按钮。
- 2) 选择“程序”→“Microsoft Visual Basic 6.0 中文版”→“Microsoft Visual Basic 6.0 中文版”，或双击 Windows 桌面上的 Visual Basic 图标。

当第一次启动 Visual Basic 时, 如果选择创建一个新的工程, 将看到如图 1-1 所示的集成开发环境界面。

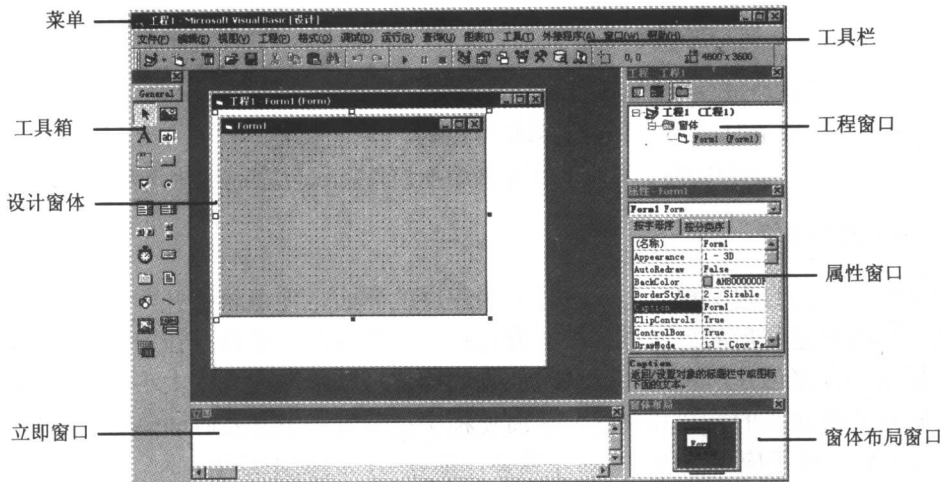


图 1-1 Visual Basic 集成开发环境

### 1.2.3 菜单和工具栏

#### 1. 菜单

Visual Basic 菜单栏包含了可以在活动窗口下使用的菜单的名字。除了提供标准“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, 例如“工程”、“格式”或“调试”。可以用“自定义”对话框中的“命令”选项卡修改菜单。

#### 2. 工具栏

工具栏在编程环境下提供对于常用命令菜单的快速访问。单击工具栏上的按钮图标, 则执行该按钮所代表的操作。例如: 标准工具栏包括打开已有工程, 保存当前工程, 剪切、复制和删除, 撤销最近的工作, 开始、暂停和结束程序的执行以及从当前环境中增加/删除窗口的图标。所有这些功能都可以通过使用下拉菜单来执行。

按照默认规定, 启动 Visual Basic 之后显示“标准”工具栏。附加的编辑、窗体设计和调试的工具栏可以从“视图”→“工具栏”菜单命令中移进或移出。

工具栏可紧贴在菜单条之下, 或以垂直条状紧贴在左边框上, 如果将它从菜单下面拖开, 则它能“悬”在窗口中。

### 1.2.4 工具箱

工具箱显示了标准的 Visual Basic 控件连同已添加到工程中的任何 ActiveX 控件和可插入对象, 例如标签、文本框、命令按钮、图片框、框架、复选框、单选按钮、文件列表框等。除了默认的工具箱布局之外, 还可以通过从上下文菜单中选定“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。要显示工具箱, 可选择“视图”→“工具箱”菜

单，或按工具栏的【工具箱】按钮。

还可以通过以下的方法来充分发挥工具箱的功能：

1) 选择“选项”对话框的“通用”选项卡中的“显示工具提示”选项，从而显示工具箱按钮的工具提示。

2) 对工具箱添加选项卡，可以定制工具箱。

3) 在添加选项卡时，指针在其上总是可用的。

4) 用“工程”菜单中的“部件”命令来添加控件，可以定制通用选项卡或专用选项卡。

## 1.2.5 设计窗体

设计窗体作为自定义窗口用来设计应用程序的界面，使得软件开发人员在程序运行以前就能看到程序运行后的界面样式。开发过程中，程序员只要直接在窗体中添加想要的控件、图形或图片来创建所希望的外观。放置的控件可以移动或改变大小等，不必编写大量代码去描述界面元素的外观和位置，从而使程序员节省许多开发时间，这也正是可视化编程的意义所在。对于应用程序中每一个窗体，都有自己的设计窗体。

在设计窗体时：

1) 每个设计窗体都有最大化、最小化和关闭按钮。

2) 可创建固定的或可移动的窗体。除非在窗体属性中另有所指，否则，设计出的窗体无论在设计时还是运行时，都将具有相同特征。

3) 用“查看”菜单中的“窗体布局窗口”命令在屏幕上预览窗体的布局。

## 1.2.6 代码窗口

代码窗口是编写应用程序的 Visual Basic 代码的地方。可以使用代码窗口来编写、显示以及编辑 Visual Basic 代码。打开各模块的代码窗口后，可以查看不同窗体或模块中的代码，并且在它们之间做复制以及粘贴的动作。代码窗口如图 1-2 所示。

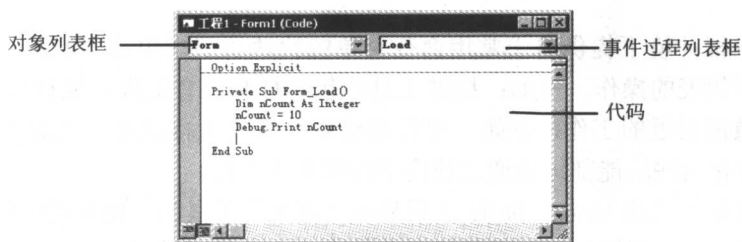


图 1-2 代码窗口

可以按照下面所述的方式打开代码窗口：

1) 在工程窗口中，可以选择一个窗体或模块，然后选择【查看代码】按钮。

2) 在“窗体”窗口中，可以双击控件或窗体，也可以从“视图”菜单中选择“代码窗口”，或者按下〈F7〉键。

代码窗口主要包括以下两个部分：

1) 对象列表框。显示所选对象的名称。可以按下列表框中右边的箭头来选择此窗体中的对象。

2) 事件过程列表框。显示对象列表框所选对象的事件过程名称。可以按下列表框的右边的箭头, 来选择对象的事件过程。此时鼠标指针会移到所选过程的第一行代码上面。

### 1.2.7 工程窗口

工程窗口显示了一个分层的文件列表, 这个列表与确定的工程相联系。从工程中创建、添加或删除可编辑文件时, 工程资源管理器窗口中会发生相应的变化。该窗口中的文件代表了各自的窗体和程序模块, 可以通过双击该窗口中的相应的图标来将一个窗体和程序模块放到工程管理器窗口中。同时, 也可以通过单击该窗口顶部最左边的两个图标之一来选择“查看对象”或“查看代码”。要显示工程资源管理器窗口, 可选择“视图”→“工程资源管理器”菜单或按〈Ctrl+R〉键。

### 1.2.8 属性窗口

Visual Basic 是面向对象的编程语言, 而每一个对象都有与之相关联的属性, 例如对象的大小、标题或颜色等。当选定窗体上的控件时, 控件的属性值将显示在属性窗口, 然后通过属性窗口修改属性值。要显示属性窗口, 可选择“视图”→“属性窗口”菜单, 或通过上下文菜单的“属性窗口”菜单, 或按〈F4〉键。

### 1.2.9 窗体布局窗口

窗体布局窗口允许使用表示屏幕的小图象来布置应用程序中各窗体的位置。如果想要改变窗体的位置, 只要简单地将窗体图标拖动到想要放置的位置就可以了。当重新调整“窗体布局窗口”时, 每个窗体都将相对于设计窗口的大小进行调整。客户区域的左上角代表桌面坐标的(0,0)。可以用快捷方式菜单上的“辨识向导”命令来改变分辨率。

### 1.2.10 Visual Basic 的对象

在 Visual Basic 中创建应用程序之际, 也是和对象打交道之时。可以使用 Visual Basic 提供的对象——譬如控件、窗体和数据访问对象。也可以在一个 Visual Basic 应用程序内部控制另一个应用程序对象。甚至还可建立自己的对象, 定义它们的属性、方法和事件。

#### 1. 对象的定义

Visual Basic 中的每个对象都是用类定义的, 类是同类对象集合的抽象, 规定了它们的公共属性和方法, 对象是类的一个实例。我们用饼干模子和饼干之间的关系作比喻, 就会明白对象和它的类之间的关系。饼干模子是类, 它确定了每块饼干的特征, 比如大小和形状。用类创建的对象就是饼干。

下面再用两个例子进一步说明 Visual Basic 中类与对象之间的关系。

(1) 在 Visual Basic 的“工具箱”上, 控件代表类, 直到在窗体上画出这些被称作控件的对象为止。它们实际上并不存在, 在创建控件之时也就是在复制控件类, 或建立控件类的实例。这个类实例就是应用程序中引用的对象。

(2) 在设计时操作的窗体是类。在运行时, Visual Basic 才建立窗体类的实例。

把对象看作是类的原原本本的复制品, 从而建立所有对象。一旦它们以单个对象的形式

式存在，属性就可改变。例如，如果在窗体上画了三个命令按钮，则每个命令按钮对象都是命令按钮类的实例。每个对象都具有一组由类定义的公共的特征和功能（属性、方法和事件）。但是，每个对象都有自己的名字，都能分别设置成有效或无效，都能放在窗体的不同位置等。

为简单起见，本章之外的大部分内容将不过多引用对象的类。例如，只要记住术语“ListBox 控件”意味着“列表框类的一个实例”。

表 1-1 列出了在 Visual Basic 中可能用到的几种类型的对象。

表 1-1 Visual Basic 中可能用到的几种类型的对象

| 示 例  | 描 述                           |
|------|-------------------------------|
| 命令按钮 | 窗体上的控件，如命令按钮和框架，它们都是对象        |
| 窗体   | Visual Basic 工程中的每一个窗体都是独立的对象 |
| 数据库  | 数据库是对象，并且还包含其他对象，如字段、索引等      |
| 图表   | Microsoft Excel 中的图表是对象       |

## 2. 对象的功能

对象可提供现成代码，省去书写的麻烦。例如，可以自己创建打开文件和保存文件的对话框。但实际上大可不必，取而代之的是利用 Visual Basic 提供的 CommonDialog 控件（一个对象）。虽然用户也能撰写日程管理和资源管理的程序，但也大可不必，也可使用 Microsoft Project 提供的 Calendar、Resources 和 Task 对象。

Visual Basic 提供了把来自不同资源的对象组合起来的工具。现在可把 Visual Basic 的各种强有力的特性以及支持自动化（先前以 OLE 自动化闻名）的应用程序结合起来，建立定制的解决方法。自动化是部件对象模式（COM）的一个特性，是应用程序使用的工业标准，用来陈列对象以开发工具和其他应用程序。

可把 Visual Basic 内部的控件结合在一起使用，也可使用其他应用程序提供的对象。考虑把下列对象放入 Visual Basic 窗体：

- 1) Microsoft Excel Chart 对象。
- 2) Microsoft Excel Worksheet 对象。
- 3) Microsoft Word Document 对象。

可用这些对象建立如图 1-3 所示的支票簿应用程序。由于不必书写代码，重建已由 Microsoft Excel 和 Word 等对象提供的功能完成，因而可节省很多时间。

### 1.2.11 对象的属性、事件和方法

Visual Basic 对象支持属性、方法和事件。

1) 属性。用于描述对象的某些外部特征。不同的对象有不同的属性，也有一些属性是共有的。用属性窗口或代码窗口可对对象的属性进行设置。

2) 方法。是对对象实施的一些动作。它实际上是对象本

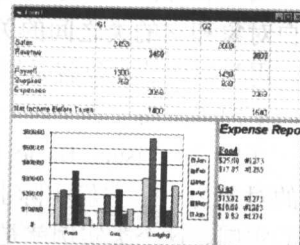


图 1-3 支票簿应用程序

身所内含的一些特殊的函数或过程，调用这些函数或过程来实现相应的动作。不同的对象具有不同的方法。

3) 事件。是由 Visual Basic 预先设置好的、能被对象识别的动作。一个对象可以识别和响应多个不同的事件。

### 1. 用属性控制对象

个别属性可以随着设置和取得它们的值的时间不同而不同。有的属性可在设计时设置，在“属性”窗口设置这些属性的值而无需编写任何代码。而有的属性在设计时是不可设置的，因此，这些属性只有通过代码在运行时设置。

在运行时可以设置并可获得值的属性叫做读写属性。在运行时只能读取的属性叫做只读属性。

### 2. 设置属性值

在想改变对象的外观或特性时，设置属性的值。例如，通过改变 TextBox 控件的 Text 属性，就可以改变文本框的内容。

用下列语法设置属性值：

```
object.property = expression
```

下面是设置属性的语句：

```
Text1.Top = 200           ' 设置 Top 属性为 200
Text1.Visible = True      ' 显示文本框
Text1.Text = "hello"      ' 在文本框中显示 "hello"
```

### 3. 读取属性值

想要在代码执行附加动作（例如给另一个对象赋值）之前得知对象的状态，就要读取属性值。例如，在运行代码之前能够返回 TextBox 控件的 Text 属性值，以确定文本框的内容。这里，代码可能改变值。

在大多数情况下可以用以下语法获得属性值：

```
variable = object.property
```

属性值可以作为较复杂的表达式的一部分，而不必将属性赋予变量。下面的代码计算控件数组中的一个新成员的 Top 属性，它等于前一个成员的 Top 属性加上 400。

```
Private Sub cmdAdd_Click ()
    ' [语句]
    optButton (n).Top = optButton (n-1).Top + 400
    ' [语句]
End Sub
```

#### 提示：

如果不只一次使用一个属性值，可将这个值存储到一个变量中，则代码执行起来会更快。