



华章教育

计 算 机 基 础 课 程 系 列 教 材

Visual Basic .NET 程序设计

李柏岩 尹枫 王洪亚 等编著



机械工业出版社
China Machine Press

计 算 机 基 础 课 程 系 列 教 材

Visual Basic .NET 程序设计

李柏岩 尹枫 王洪亚 周余洪 杜明 编著



机械工业出版社
China Machine Press

本书详尽地介绍了 Visual Basic .NET 的基础知识和程序设计方法以及使用 Visual Basic .NET 绘图和进行数据库开发的方法。全书共两部分，第一部分（第 1~10 章）为原理篇，主要内容包括：VB.NET 集成开发环境、面向对象程序设计的概念、窗体和基本控件、Visual Basic .NET 语言基础和常用语句、数组、函数、过程、界面设计、绘图、文件和数据库等。第二部分（实验一~实验十）为实验篇，给出了与原理篇各章相对应的 10 个实验。

本书内容丰富、注重应用，既可作为高等院校非计算机专业学生的教材和教学参考书，也可作为广大读者的自学用书。

封底无防伪标均为盗版

版权所有，侵权必究

本书法律顾问 北京市展达律师事务所

图书在版编目（CIP）数据

Visual Basic .NET 程序设计/李柏岩等编著. —北京：机械工业出版社，2011.8

（计算机基础课程系列教材）

ISBN 978-7-111-35415-4

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

中国版本图书馆 CIP 数据核字（2011）第 148788 号

机械工业出版社（北京市西城区百万庄大街 22 号 邮政编码 100037）

责任编辑：王 璐

北京市荣盛彩色印刷有限公司印刷

2011 年 9 月第 1 版第 1 次印刷

185mm×260mm·13.75 印张

标准书号：ISBN 978-7-111-35415-4

定价：28.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

客服热线：(010) 88378991；88361066

购书热线：(010) 68326294；88379649；68995259

投稿热线：(010) 88379604

读者信箱：hzjsj@hzbook.com

前 言

Visual Basic .NET (简称 VB.NET) 是一种可视化的面向对象程序设计语言, 由于其功能强大、容易掌握、使用方便, 深受广大程序设计工作者的喜爱, 成为目前许多专业人员和编程爱好者不可缺少的编程工具。

本书主要针对已经具备计算机基础知识的非计算机专业学生, 围绕“怎样设计 VB.NET 程序解决实际问题”这个中心, 通过大量有实际意义的程序实例, 深入浅出地介绍了 VB.NET 程序设计的基本概念、方法和技术。本书内容丰富、实例生动, 重点突出、注重实用, 文字流畅, 便于读者自学。

全书共分两部分, 第一部分 (第 1 ~ 10 章) 为原理篇, 其中, 第 1 章介绍了 VB.NET 的历史发展轨迹、与 Microsoft .NET 的关系及 VB.NET 集成开发环境; 第 2 章简要介绍了面向对象的基本概念, 并自然过渡到 VB.NET 程序开发机制, 介绍了窗体和基本控件及其使用; 第 3 章介绍了 VB.NET 语言方面的基本知识; 第 4 章讲解了 VB.NET 程序设计的基本语句; 第 5 章讲解了数组及其应用; 第 6 章介绍了自定义函数和过程; 第 7 章介绍了用户界面设计的相关知识; 第 8 章介绍 VB.NET 的文件系统及其使用; 第 9 章介绍了 VB.NET 的 CDI + 绘图功能; 第 10 章介绍了数据库开发基础。每一章后面都设置了习题, 用于测试学生对所学知识点的情况。第二部分 (实验一 ~ 实验十) 为实验篇, 安排了与第一部分相对应的 10 个配套实验, 并对实验中的难点给出了相应的提示。

此外, 本书的例题、实验题都在 Visual Basic .NET 环境中调试通过。

本书的编写人员大多参与编写了强莎莎老师 2007 年主编的《Visual Basic .NET 程序设计》一书, 从中获得了很多宝贵经验。本书也借鉴了该书的一些思路和写法, 并获得强莎莎老师和骆轶姝老师的支持, 在此表示感谢。

参加本书编写的有李柏岩、尹枫、王洪亚、周余洪、杜明。本书编写过程中, 得到了刘晓强、宋晖老师的大力帮助和指导, 在此表示深深的谢意。

限于作者水平, 书中有不当之处, 欢迎广大读者批评指正。

编者

2011 年 6 月

教学建议

教学章节	教学要求	课时
第 1 章 Visual Basic .NET 概述	了解 BASIC、Visual Basic .NET、Microsoft .NET 的特点及内在联系	1
	了解 Visual Basic .NET 集成开发环境的安装、界面及基本功能 掌握开发 VB.NET 应用程序的方法	2
	了解 VB.NET 常用调试工具, 掌握程序调试方法 掌握 VB.NET 帮助系统的使用方法	1
第 2 章 VB.NET 程序设计初步	了解面向对象程序设计的基本概念 掌握 VB.NET 事件驱动的编程机制	1
	掌握 VB.NET 窗体及常用控件的属性、方法、事件及其使用方式	6
第 3 章 VB.NET 语言基础	掌握 VB.NET 基本数据类型, 包括常量、变量等	1
	掌握 VB.NET 常用运算符、表达式的使用	2
	掌握 VB.NET 常用函数及其使用	2
第 4 章 基本的程序结构	了解结构化程序设计的基本结构: 顺序结构、选择结构、循环结构 掌握主要程序语句, 包括赋值语句、选择语句和循环语句等	14
第 5 章 数 组	了解数组的基本概念和用法 掌握数组的声明、重新声明及操作方法 掌握数组元素的查询、排序算法 掌握数组相关控件及结构数组的用法	8
第 6 章 自定义函数和过程	掌握函数、过程的定义和调用方法 掌握参数传递的原理和用法	8
第 7 章 用户界面设计	掌握菜单栏、工具栏、状态栏的用法 掌握鼠标、键盘的编程方法	6
第 8 章 文 件 系 统	了解文件的基本概念 掌握顺序文件、随机文件的读写方法 了解目录管理和文件管理的方法	8
第 9 章 绘 图	了解 GDI + 图形系统 掌握基本的绘图对象及绘图语句的用法	4
第 10 章 (可选) 数据库访问	了解数据库访问的基本方法 掌握访问数据库的方法和步骤	4
第 1 ~ 10 章建议课时		64 ~ 68
综合实例 (第 2 ~ 9 章) 建议课时		20 ~ 30

说明:

- 1) 建议课堂教学全部在多媒体机房内完成, 实现“讲-练”结合。
- 2) 建议课时包括上课和实验两部分, 在时间分配上大致各占一半。
- 3) 建议教学分为核心知识技能模块 (第 1 ~ 10 章的基本内容) 和扩展提高模块 (第 2 ~ 9 章的综合实例), 其中核心知识技能模块建议教学学时为 64 ~ 68, 技能提高模块建议学时为 20 ~ 30, 不同学校可以根据各自的教学要求和计划学时数对教学内容进行取舍。

目 录

前言

教学建议

第一部分 原 理 篇

第 1 章 Visual Basic.NET 概述	1
1.1 VB.NET 简介	1
1.1.1 VB.NET 语言的由来	1
1.1.2 Microsoft.NET 框架	1
1.1.3 VB.NET 的新特点	3
1.2 VB.NET 集成开发环境	3
1.2.1 VVB.NET 安装和启动	3
1.2.2 VB.NET 开发环境界面	5
1.3 创建 VB.NET 应用程序	8
1.4 程序调试	11
1.4.1 程序错误	11
1.4.2 程序的调试	12
1.5 VB.NET 帮助系统	14
1.5.1 安装 MSDN Library	14
1.5.2 使用 MSDN Library	15
习题	16
第 2 章 VB.NET 程序设计初步	18
2.1 面向对象程序设计的基本概念	18
2.1.1 对象和类的概念	18
2.1.2 VB.NET 的类和命名空间	20
2.1.3 VB.NET 的编程机制	21
2.2 窗体和基本控件	22
2.2.1 控件的基本属性	23
2.2.2 窗体	24
2.2.3 按钮	27
2.2.4 标签和文本框	28
2.2.5 图片框控件	31
2.2.6 定时器、时间控件和滚动条	32
2.3 综合实例	34
2.3.1 问题及分析	34
2.3.2 相关知识	35
2.3.3 解决方案	37
2.3.4 实现代码	38
习题	39

第 3 章 VB.NET 语言基础	41
3.1 基本数据类型	41
3.1.1 引例	41
3.1.2 数据类型	42
3.2 常量与变量	43
3.2.1 标识符的命名规则	43
3.2.2 常量	43
3.2.3 变量	45
3.3 运算符和表达式	46
3.3.1 运算符	46
3.3.2 表达式	49
3.4 常用函数	50
3.4.1 数学函数	51
3.4.2 字符串函数	53
3.4.3 日期和时间函数	55
3.4.4 输入函数与提示函数	57
3.4.5 Shell 函数	59
3.5 综合实例	60
3.5.1 问题及分析	60
3.5.2 相关知识	60
3.5.3 解决方案	61
3.5.4 实现代码	62
习题	63
第 4 章 基本的程序结构	64
4.1 顺序结构	64
4.1.1 赋值语句	64
4.1.2 程序结束语句	66
4.2 选择结构	66
4.2.1 If 语句	66
4.2.2 Select Case 语句	69
4.2.3 Goto 语句	71
4.2.4 选择和分组控件	72
4.3 循环结构语句	74
4.3.1 计数循环控制语句	75
4.3.2 条件循环控制语句	77
4.4 综合实例	79
4.4.1 问题及分析	79
4.4.2 解决方案	80

4.4.3 实现代码	81	7.1 菜单、工具栏和状态栏	122
习题	82	7.1.1 引例	122
第5章 数组	84	7.1.2 菜单	123
5.1 数组的概念	84	7.1.3 工具栏	126
5.2 数组的声明	84	7.1.4 状态栏	128
5.2.1 数组声明	84	7.2 窗体和对话框	129
5.2.2 数组初始化	85	7.2.1 多窗体界面	129
5.2.3 重新声明数组	86	7.2.2 通用对话框	132
5.3 数组的访问	87	7.3 键盘和鼠标	133
5.3.1 数组的基本操作	87	7.3.1 键盘	133
5.3.2 数组的排序与快速查找	89	7.3.2 鼠标	135
5.4 列表框和组合框	92	7.4 综合实例	136
5.4.1 列表框	92	7.4.1 问题及分析	137
5.4.2 组合框	93	7.4.2 解决方案	137
5.5 结构类型及结构数组	94	7.4.3 实现代码	140
5.5.1 结构类型	94	习题	142
5.5.2 结构数组	95	第8章 文件系统	144
5.6 综合实例	96	8.1 文件的概念	144
5.6.1 问题及分析	96	8.1.1 文件	144
5.6.2 相关知识	97	8.1.2 VB.NET 文件分类	144
5.6.3 解决方案	98	8.1.3 VB.NET 的文件访问	145
5.6.4 实现代码	99	8.2 顺序文件	145
习题	103	8.2.1 基本操作	145
第6章 自定义函数和过程	104	8.2.2 顺序文件的使用	148
6.1 函数过程	104	8.3 随机文件	151
6.1.1 内部函数	104	8.3.1 基本操作	151
6.1.2 自定义函数的定义	104	8.3.2 随机文件的使用	152
6.2 子过程	107	8.4 VB.NET 文件管理	154
6.2.1 子过程的定义	107	8.4.1 目录管理	154
6.2.2 子过程的调用	107	8.4.2 文件管理	156
6.3 参数传递	110	8.5 综合实例	157
6.3.1 形参和实参	110	8.5.1 问题及分析	157
6.3.2 传值与传地址	110	8.5.2 解决方案	157
6.4 模块与作用域	112	8.5.3 实现代码	158
6.4.1 标准模块	112	习题	159
6.4.2 变量作用域	114	第9章 绘图	161
6.4.3 过程作用域	115	9.1 GDI + 图形系统简介	161
6.5 综合实例	115	9.2 GDI + 绘图基础	161
6.5.1 问题及分析	115	9.2.1 GDI + 坐标系	161
6.5.2 相关知识	116	9.2.2 GDI + 绘图基本步骤	161
6.5.3 解决方案	117	9.2.3 GDI + 的常用结构和类	162
6.5.4 实现代码	119	9.3 图形绘制	166
习题	120	9.3.1 绘制图形	166
第7章 用户界面设计	122	9.3.2 填充图形	166

9.3.3 绘制文字	167
9.3.4 绘制图像	167
9.4 综合实例	167
9.4.1 问题及分析	167
9.4.2 解决方案	168
9.4.3 实现代码	168
习题	169
第 10 章 数据库访问	170
10.1 创建数据库	170
10.1.1 新建 Access 数据库	170
10.1.2 数据库基本概念	172
10.2 结构化查询语言 SQL	173
10.2.1 SQL 语言介绍	173
10.2.2 SQL 语言分类	173
10.2.3 SQL 语言应用	173
10.3 数据库访问基础	177
10.3.1 在 VB.NET 平台中连接和操作 数据库	178
10.3.2 ADO.NET 简介	179

10.3.3 ADO.NET 对象模型	180
10.4 基于绑定的数据库访问	182
10.4.1 使用数据控件访问数据库	182
10.4.2 数据库操作编程	186
习题	187

第二部分 实验篇

实验一 VB.NET 集成开发环境	188
实验二 VB.NET 常用基本控件	189
实验三 VB.NET 语言基础	192
实验四 基本流程控制	194
实验五 数组	199
实验六 自定义函数与过程	202
实验七 用户界面设计	205
实验八 文件系统	207
实验九 绘图	207
实验十 数据库访问	208
参考文献	210

第一部分 原 理 篇

第 1 章 Visual Basic .NET 概述

Visual Basic .NET 简称 VB.NET，是一种面向对象的高级程序设计语言及编程环境，是 Microsoft .NET 平台的重要组成部分。本章主要介绍 VB.NET 集成开发环境及其使用。通过本章的学习，读者可以了解 VB.NET 语言的发展历史、基本特性及集成开发环境，学会使用集成开发环境创建、运行和调试 VB.NET 应用程序的基本方法和步骤。

1.1 VB.NET 简介

VB.NET 程序设计语言由最初的 BASIC 语言发展而来，并随着相关技术平台的发展而不断演变，但一直保持简单易学、功能强大的特点，是计算机技术发展史上应用最为广泛的程序设计语言之一。

1.1.1 VB.NET 语言的由来

BASIC 语言是美国 Dartmouth 学院的 John G. Kemeny 与 Thomas E. Kurtz 两位教授在 20 世纪 60 年代中期提出的一种高级程序设计语言，其英文全称为“Beginner's All-Purpose Symbolic Instruction Code”。BASIC 语言定位为“适用于初学者的多功能符号指令代码”，最初是一种解释式语言，即所谓“边解释，边执行”，后来逐渐发展为兼具解释与编译双重翻译方式的语言。由于 BASIC 语言易学易用，很快就流行起来了。

BASIC 语言在发展过程中出现许多版本，如 BASICA、Quick BASIC 和 GWBASIC 等。虽然这些版本在语法、规则、功能等方面并不完全相同，但几乎都继承了 BASIC 创始者所设计的基本形态与精神。其中最具代表性、应用最广的是微软公司在 1991 年首次推出的 Visual Basic 系列开发环境。Visual Basic 支持“所见即所得”的可视化编程模式，以结构化 BASIC 语言为基础，以事件驱动为运行机制，使用窗体、可视化控件和代码设计程序。

Visual Basic 不仅能开发独立的 Windows 应用程序，还能以脚本方式嵌入 Microsoft 办公系列产品 Word、Excel 和 PowerPoint 等软件，为用户提供编程接口，扩展这些软件的功能。这种脚本语言称为 Visual Basic for Applications（简称 VBA），也称宏语言，是 Visual Basic 语言的子集。另一种常用的 Visual Basic 脚本语言是 VBScript，简称 VBS。VBS 作为活动服务器页（Active Server Page，ASP）技术的默认语言，主要用于动态网页的设计。

Visual Basic 的版本从 1.0 发展到 6.0 之后，Microsoft 将其整合到一个称为“Microsoft .NET”（简称 .NET）的全新技术框架中，形成了 Visual Basic .NET。

1.1.2 Microsoft .NET 框架

Microsoft .NET 是 Microsoft 公司于 2000 年 6 月推出的一个技术框架（Framework），目的是将

已有的技术整合起来,建立一个用于创建、部署和运行新一代基于 Internet 应用程序的统一环境。这个框架包括两个主要组成部分:公共语言运行时 (Common Language Runtime, CLR) 和 .NET 基础类库 (Base Class Library, BCL)。

1. 公共语言运行时

CLR 像 Java 虚拟机一样也是一个运行时环境,是 .NET 框架的核心和基石,它负责资源管理并监控程序的运行,保证应用与底层操作系统之间必要的分离。基于 CLR 语言编译器开发的代码称为托管 (Managed) 代码,否则称为非托管代码。托管代码具有许多优点,例如:跨语言集成、跨语言异常处理、增强的安全性、版本控制和部署支持等。凡是符合公共语言规范 (Common Language Specification) 的程序语言所开发的程序都可以在 CLR 上运行。有了 CLR,开发人员就能容易地设计出能够跨语言交互的组件和应用程序。

2. .NET 基础类库

.NET 框架的类库是一个与 CLR 紧密集成的可重用类集合,它提供的功能涉及从输入、输出到数据访问等多个环节的内容,为开发人员提供了一个统一的、面向对象的、层次化的可扩展应用程序编程接口 (API)。.NET 基础类库由 2500 多个可重用类构成,.NET 框架支持的所有语言都可以使用同一个基础类库开发各种应用程序,而在 .NET 出现之前的传统的语言编程环境中,不同的语言需要有不同的函数库,使用不同的调用方法。

此外,活动服务器页 (ASP+) 提供了基于统一类库的 Web 应用程序模型。该模型由一个基本结构和一组可运行在 Web 服务器上的控件构成。使用该模型,开发人员可以方便地构建各种 Web 应用。Microsoft .NET 中访问数据库的技术是 ADO.NET。ADO.NET 提供了一组用来连接数据库、运行命令、返回记录集的类。XML 技术则作为一种通用数据交换格式,奠定了新一代数据交换的标准。

Microsoft .NET 框架、开发工具和编程语言共同构成了 Microsoft .NET 统一开发平台,其结构如图 1-1 所示。



图 1-1 Microsoft .NET 开发平台

.NET 框架是整个开发平台的基础,而集成开发工具 Visual Studio .NET 则是平台最重要的开发工具,它支持 Visual Basic、Visual C++、Visual C#、Visual J# 等多种编程语言。在 Visual Studio .NET 中,用户可以使用其中任何一种语言环境创建应用程序,而且很容易将其他语言编写的模块整合在一起。在 Microsoft .NET 平台中,不同的编程语言编写的程序首先被翻译成统一的中间语言代码 (MSIL),然后在 CLR 的支持下,中间语言程序被编译成由本地 CPU 指令组成的可执行程序。这些程序使用同一个基础类库,能够访问相同的底层对象和组件,在语言层次上

已没有执行效率上的差别了。

1.1.3 VB.NET 的新特点

VB.NET 是一种面向对象的编程语言,虽然它是从 Visual Basic 语言发展而来的,但不是对后者的简单改进和升级。VB.NET 与 Visual Basic 最大的差别是引入了 .NET 框架,.NET 框架为 VB.NET 编程提供了一个庞大的基础类库,构成了 VB.NET 的编程基础。

VB.NET 沿用了 Visual Basic 语言的多数语法,并做了少量改进,以适应 .NET 框架对编程语言的要求,但同样具有简单易用的特点。VB.NET 包含了 Visual Basic 6.0 的所有控件,为大多数控件增添了新的属性、方法和事件,而且还增加了新的控件和组件用于创建用户界面。另外,VB.NET 继承了 Visual Basic 的可视化设计和事件驱动的编程机制,具有丰富的数据类型和完备的帮助功能,支持强大的数据库访问和网络功能,为快速创建 Windows 应用程序提供了一种有效途径。

1.2 VB.NET 集成开发环境

Visual Basic .NET 集成开发环境 (Integrated Development Environment, IDE) 是 Microsoft Visual Studio .NET 的一部分,它集程序代码的编辑、编译、连接和调试等功能于一体,给编程人员提供了一个完整、方便的开发界面和许多有效的开发辅助工具。本书将以 Microsoft 公司 2005 年推出的、可运行在 Windows XP 操作系统上的 Visual Studio .NET 2005 为背景,全面介绍 VB.NET 编程语言及其集成开发环境。

1.2.1 VB.NET 安装和启动

Microsoft Visual Studio .NET 2005 的安装非常简单,既可以通过光驱安装,也可将安装程序复制到硬盘上安装。本书以硬盘安装为例,打开如图 1-2 所示的安装文件夹,双击 SETUP.EXE 文件图标,就可以启动安装程序。

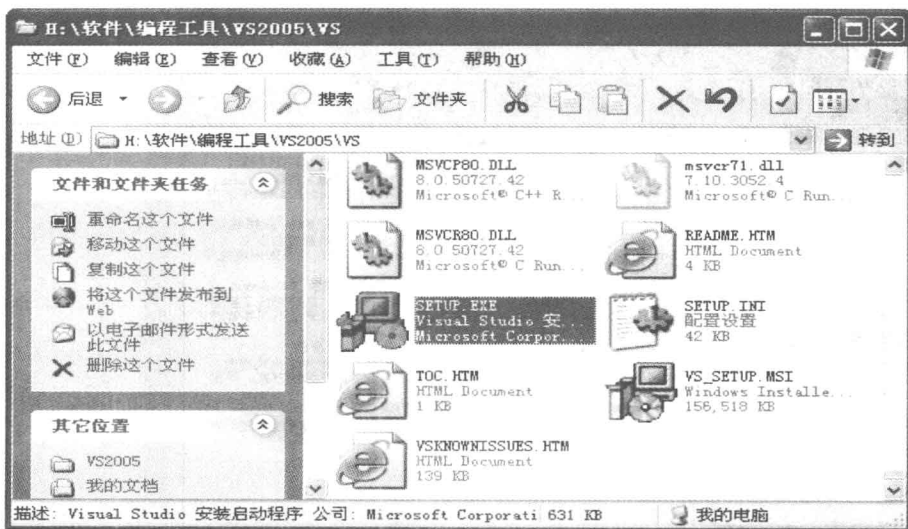


图 1-2 Visual Studio 2005 安装文件夹

Visual Studio 2005 的安装程序界面如图 1-3 所示,点击“安装 Visual Studio 2005”选项后,安装向导推动安装过程自动进行。Visual Studio 2005 提供了三个安装选项:①“默认”选项只

安装最常见的功能（约占 2GB 存储空间）；②“完全”选项将安装所有功能（约占 4GB 存储空间）；③“自定义”选项允许用户去掉一些不需要的功能。若选择“默认”安装，后续的过程几乎不需要用户干预。

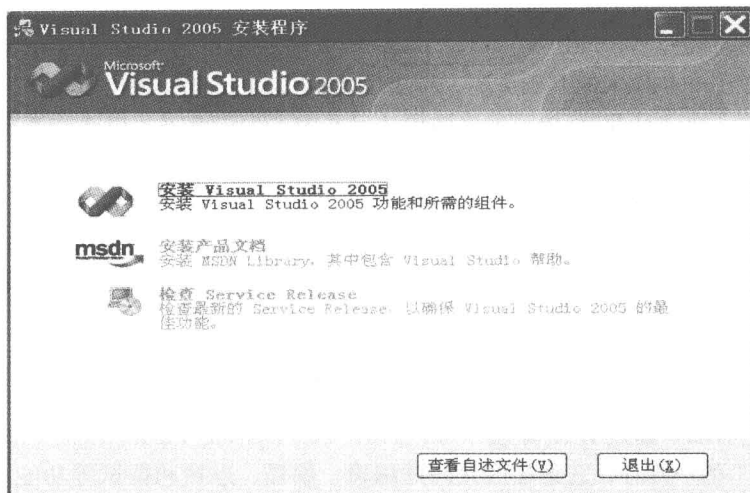


图 1-3 Visual Studio 2005 安装程序界面

安装完成后，Windows 的“程序”菜单中会出现“Microsoft Visual Studio 2005”菜单项，点击该项即可启动 Visual Studio 2005 集成开发环境。

Microsoft Visual Studio 2005 支持多种程序设计语言，第一次启动时会提示用户选择开发语言，如果用户选择 Visual Basic .NET，则系统启动后的界面如图 1-4 所示。

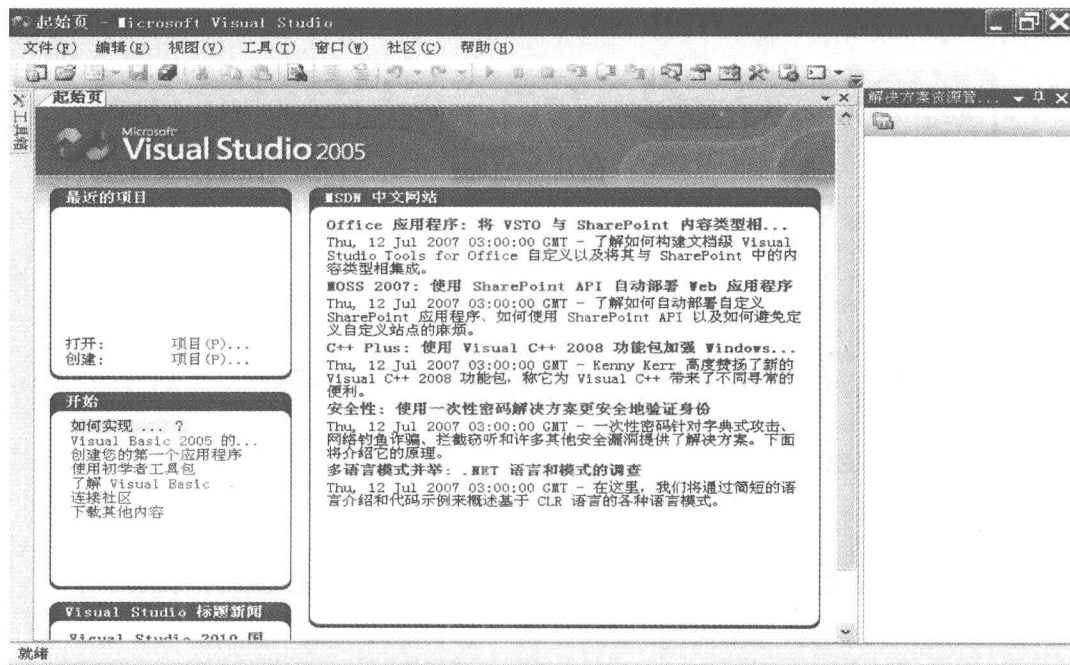


图 1-4 Visual Studio 2005 的初始界面

1.2.2 VB.NET 开发环境界面

VB.NET 集成开发环境是 Microsoft Visual Studio 的一部分，其界面主要由标题栏、菜单栏、工具栏、工具箱、解决方案资源管理器、属性窗口、设计窗口、代码窗口、错误列表窗口和状态栏等组成，其中大多数窗口都是浮动和可关闭的，当窗体布局被改变后，想重新得到默认窗口布局，可执行“窗口 | 重置窗口布局”菜单命令。新建一个 Visual Basic .NET 项目后，设计界面如图 1-5 所示。

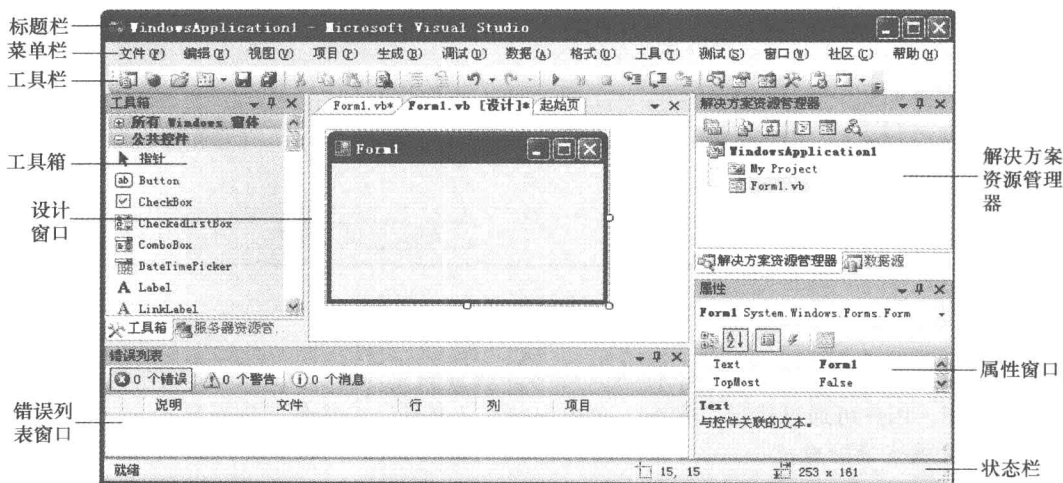


图 1-5 Visual Studio 2005 的设计界面

1. 标题栏、菜单栏和工具栏

VB.NET 开发环境的标题栏、菜单栏和工具栏如图 1-6 所示。标题栏用于显示当前项目的名称和状态。菜单栏包括 13 个菜单，其中“文件”菜单提供新建、打开和保存项目的功能；“编辑”菜单提供代码编辑操作的命令；“视图”菜单用于管理工作区中显示的窗口；“项目”菜单提供项目管理功能；“生成”菜单用于产生程序运行代码；“调试”菜单提供调试程序的功能。工具栏通过按钮为常用功能提供快速启动方式。

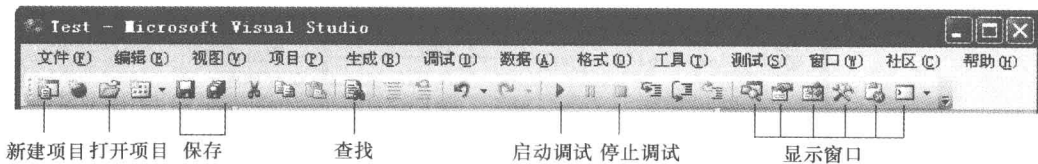


图 1-6 集成开发环境的标题栏、菜单栏和工具栏

2. 工具箱

工具箱以图标和名称列表的形式提供控件，用于设计程序界面。工具箱中只显示部分常用控件，这些控件按类型分成多个组。事实上，用户可以通过右键菜单向工具箱中添加选项卡和控件。例如，要添加其他控件到工具箱，可在控件列表的右键菜单中执行“选择项”命令，在弹出的如图 1-7 所示的对话框中找到所需控件，选中控件名称左边的复选框，单击“确定”按钮即可。

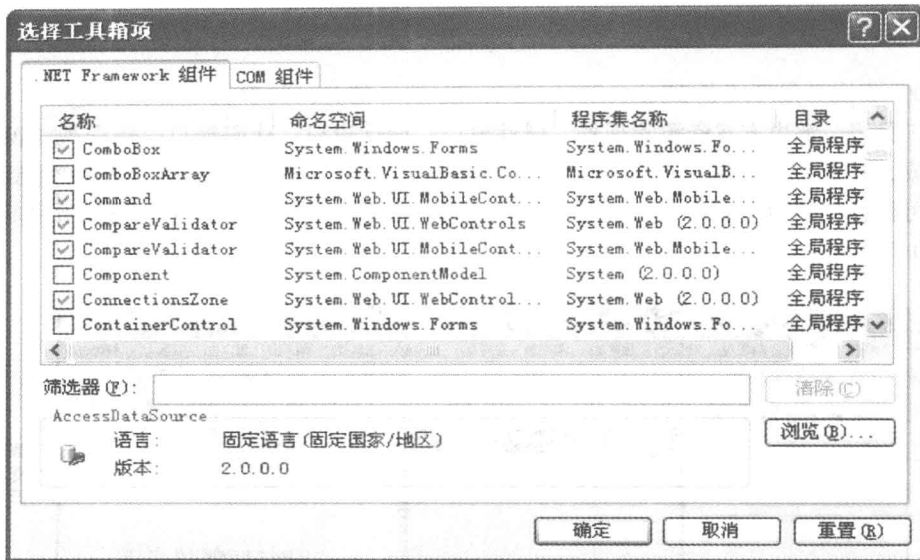


图 1-7 “选择工具箱项”对话框

要使用工具箱中的控件，可用鼠标将其拖动并放置到窗体上。工具箱也是一个浮动窗口，若窗口被关闭，用户可通过执行“视图”菜单中的“工具箱”命令使其重新显示。

3. 解决方案资源管理器

VB.NET 提供解决方案和项目两种容器来管理开发工作所需要的项，如引用、数据连接、文件夹和文件等。一个解决方案可以包含多个项目，而一个项目通常包含多个项。解决方案资源管理器以树形结构组织 VB.NET 项目中的所有相关项，如图 1-8 所示。用户可以使用解决方案资源管理器管理项目资源，例如，要打开设计窗口，用鼠标双击管理器中的窗体文件名即可。用户可以通过单击“显示所有文件”按钮查看更多的文件夹和文件；通过“查看代码”和“查看设计器”按钮切换设计窗口和代码窗口。

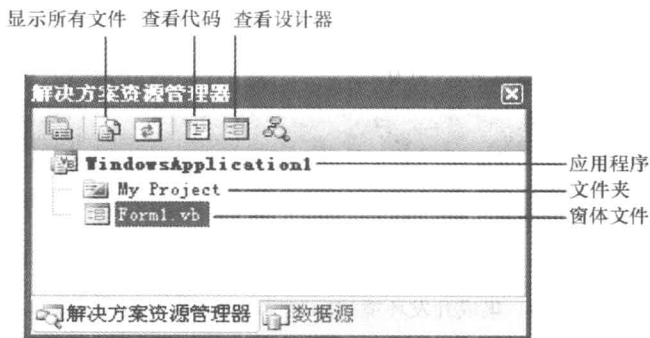


图 1-8 解决方案资源管理器

4. 属性窗口

属性窗口显示选中对象的属性，如图 1-9 所示。每个对象都由一组属性来描述其特征，如名称、颜色、大小等。在程序设计阶段，用户可以通过属性窗口查看和修改对象的属性值和事件关联信息。属性和事件既可以按分类顺序显示，也可以按字母顺序显示。“属性”和“事件”按钮用于切换属性和事件的显示。



图 1-9 属性窗口

5. 设计窗口

设计窗口是进行程序界面设计的主要窗口，如图 1-10 所示。窗体是可视化控件的主要容器，用户可以在其上添加各种控件。程序运行时，通常以窗体为界面形式，用户通过与窗体上可视控件进行交互来使用程序。一个应用程序可以有多个窗体，用户可通过选择“项目 | 添加 Windows 窗体”菜单项添加新的窗体。

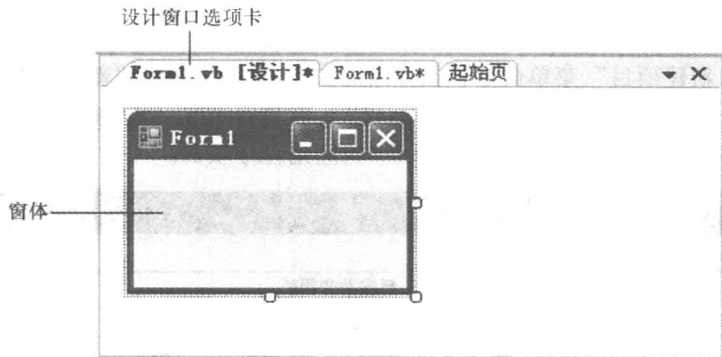


图 1-10 设计窗口

6. 代码窗口

代码窗口是用户输入程序代码的主窗口，如图 1-11 所示。用户可通过单击代码窗口上方的选项卡或双击窗体、控件进入代码窗口，并在窗口中输入、编辑、调试程序代码。代码窗口左上部的组合框可以显示窗体或控件的类名，右上部的组合框显示该类对应的方法名称。类的每一个方法都对应该类可能发生的一个事件，当用户选择一个类的方法后，系统会在代码窗口自动创建该类相关事件过程的首部和尾部，用户可在其中间添加程序代码，来实现该事件发生时程序应完成的功能。

7. 错误列表窗口

错误列表窗口是代码编辑过程中非常重要的错误检查窗口，通常显示在代码编辑窗口下方。错误列表列出了当前应用程序的所有语法错误，并且指明错误所在的行号、列号和文件。双击错误列表中的错误项，可以直接定位到代码窗口中的错误代码上。

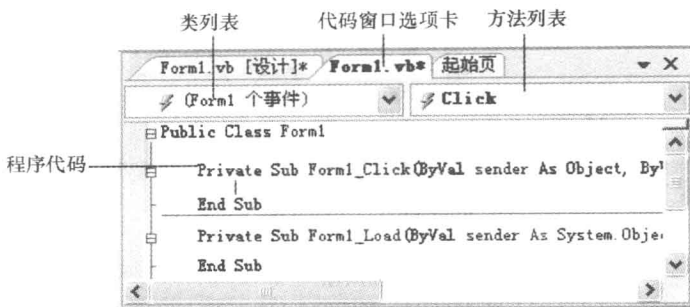


图 1-11 代码窗口

错误列表窗口用于显示程序的错误信息，将在 1.4.1 节进行介绍；状态栏用于显示系统信息和用户的提示信息，将在 7.1.4 节详细介绍。

1.3 创建 VB.NET 应用程序

VB.NET 集成开发环境有设计、运行和调试三种工作模式。在设计模式下，用户可进行界面设计和代码编写。应用程序运行时，系统处于运行模式下（标题栏显示“正在运行”），这时不能编辑代码，也不能修改界面。当应用程序出错或被中断时，系统处于调试模式（标题栏显示“正在调试”），这时可编辑代码，调试程序。本节将通过新建一个名称为 Test 的 VB.NET 项目，学习创建 VB.NET 程序的基本方法和步骤。

1. 创建 VB.NET 项目

执行“文件 | 新建项目”菜单命令，在弹出的如图 1-12 所示的“新建项目”对话框中，选择“Visual Basic”项目，并在模板中选择“Windows 应用程序”。输入项目名称“Test”后单击“确定”按钮，VB.NET 将创建一个名为“Test”的应用程序项目。

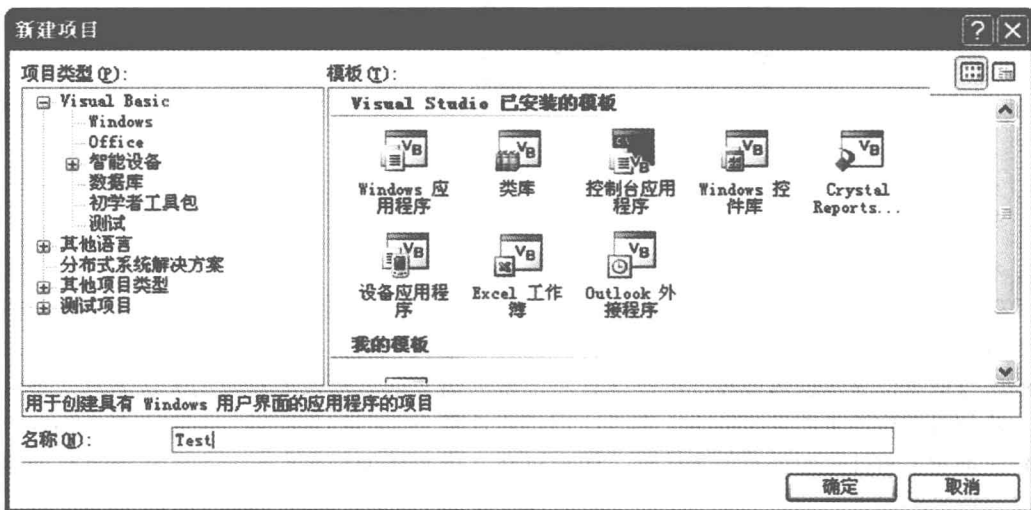
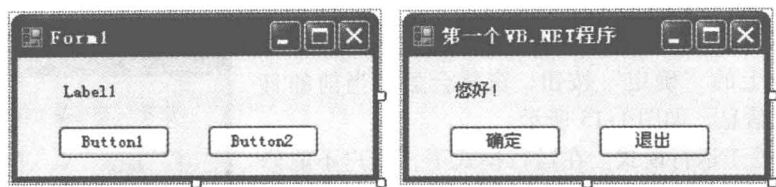


图 1-12 “新建项目”对话框

此时 IDE 处于设计工作模式，设计窗口中显示一个名称为 Form1 的空白窗体，用户可通过向窗体添加控件来设计程序界面。

2. 设计程序界面

在工作区中选中窗体 Form1，用鼠标拖动窗体四周的白色小方块来将窗体调整到合适大小。从工具箱中拖拽一个 Label 控件、两个 Button 控件到窗体上，按图 1-13a 所示布局放置。



a) 拖放控件后的窗体

b) 修改控件属性后的窗体

图 1-13 添加控件后的窗体

选中窗体或窗体上的控件，按表 1-1 在属性窗口中修改控件的属性值。修改后的窗体如图 1-13b 所示。

表 1-1 控件属性值

控件名称	属性名称	属性值
Form1	Text	第一个 VB.NET 程序
Label1	Text	您好!
Button1	Text	确定
Button2	Text	退出

Test 程序的界面设计完成后，还需要编写程序代码，使程序具有一定的功能。

3. 编写程序代码

在设计窗口中用鼠标双击窗体上的“确定”按钮，则工作区中显示代码窗口。在窗口的已出现的 Button1_Click 事件过程中添加如下程序代码：

```
Label1.Text = "您好!,现在是 " + Now()
```

选择工作区上方的“Form1.vb[设计]*”选项卡，重新进入设计窗口，双击“退出”按钮，在 Button2_Click 事件过程中添加如下程序代码：

```
End
```

上述添加的两行代码中，第 1 行的含义是将“您好!，现在是”字符串和 Now 函数返回的当前时间连接起来，赋值给 Label1 控件的 Text 属性，用以在窗体上显示；第 2 行的含义是结束当前程序的运行。最后，代码窗口的内容如图 1-14 所示。

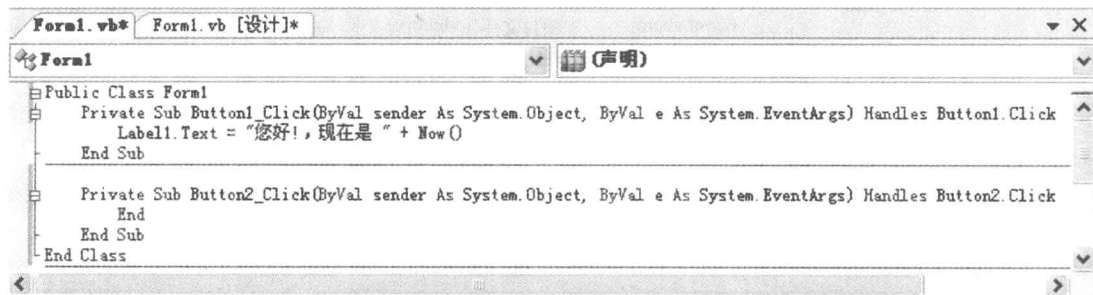


图 1-14 代码窗口中的程序代码