



21世纪高等学校计算机
专业实用规划教材

Visual Basic.NET 经典游戏开发

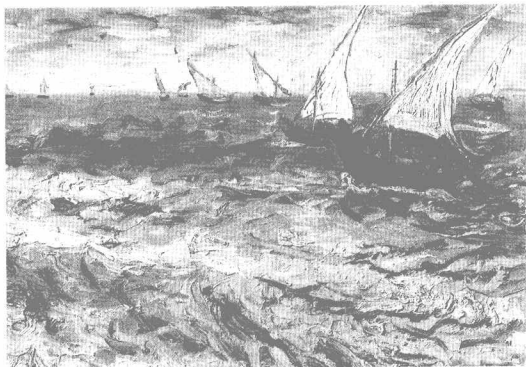
◎ 夏敏捷 樊银亭 主编
赵一丁 张秋闻 张西广 副主编



清华大学出版社



21世纪高等学校计算机
专业实用规划教材



Visual Basic.NET经典游戏开发

◎ 夏敏捷 樊银亭 主编
赵一丁 张秋闻 张西广 副主编

清华大学出版社
北京

内 容 简 介

本书以游戏实例的形式向读者展示了 Visual Basic.NET 的编程精髓, 将 Visual Basic.NET 编程中语法、技巧等方面的知识融会于其中。书中涉及的游戏都是大家耳熟能详的, 如连连看、推箱子、俄罗斯方块、象棋、坦克大战等, 使枯燥的语言学习充满乐趣。本书在讲解过程中不仅列出了完整的游戏代码, 而且对所有的源代码进行了非常详细的解释, 做到了通俗易懂, 图文并茂。恰当的游戏实例特别有助于初学者仿效理解、把握问题的精髓, 在实践中掌握和领会 Visual Basic.NET 编程的基本技巧和思想精髓。

本书既适合初学者和具有一定编程经验的 Visual Basic 和 Visual Basic.NET 用户学习, 也可供广大计算机工作者、游戏编程爱好者和软件开发人员参考。

本书封面贴有清华大学出版社防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目 (CIP) 数据

Visual Basic.NET 经典游戏开发/夏敏捷, 樊银亭主编. --北京: 清华大学出版社, 2016
ISBN 978-7-302-44170-0

I. ①V… II. ①夏… ②樊… III. ①游戏程序—BASIC 语言—程序设计 IV. ①TP311.5

中国版本图书馆 CIP 数据核字 (2016) 第 148597 号

责任编辑: 刘 星 王冰飞

封面设计: 刘 键

责任校对: 胡伟民

责任印制: 何 芊

出版发行: 清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址: 北京清华大学学研大厦 A 座 邮 编: 100084

社 总 机: 010-62770175 邮 购: 010-62786544

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

课 件 下 载: <http://www.tup.com.cn>, 010-62795954

印 装 者: 三河市春园印刷有限公司

经 销: 全国新华书店

开 本: 185mm×260mm 印 张: 23.25 字 数: 585 千字

版 次: 2016 年 9 月第 1 版 印 次: 2016 年 9 月第 1 次印刷

印 数: 1~2000

定 价: 49.00 元

前言

为什么学习 Visual Basic.NET?

微软公司的.NET战略是一场软件革命,它改变了开发人员开发应用程序的方式及思维,使开发人员能创建出全新的应用程序,这些应用程序是在.NET框架下实现的。Visual Basic.NET 程序结构十分清晰,较易学习和使用,同时又不失灵活性和强大的功能,在开发能力和效率之间取得了较好的平衡。它不仅具有快速开发应用程序的能力,而且具有 C++ 的基本特征——面向对象,Visual Basic.NET 已成为功能强大的面向对象的编程语言。

在.NET框架中,VC++.NET 已经退化成底层的语言了。就像 C++ 之前的标准 C 语言一样,它虽然仍旧强大,但已经不是主流,以后只有开发底层或桌面程序的程序员使用它。而 Visual Basic.NET 是.NET框架的支柱。Visual Basic.NET 可以充分利用.NET框架的类库和公共语言运行库编写出功能强大的各种应用程序。作为.NET框架下的核心技术之一,经过几年的发展,Visual Basic.NET 已经成为主流的开发语言。

本书内容丰富、全面,包含 21 种游戏的开发实例,涵盖了益智、射击、棋牌、休闲、网络等游戏类型。其中的通用代码可直接应用于一般的游戏。每个游戏实例均提供了详细的设计思路、关键技术分析以及具体的解决方案。这样每个游戏实例都是活的、实用的 Visual Basic.NET 编程实例。

需要说明的是,学习编程是一个实践的过程,而不仅是看书、看资料的过程,亲自动手编写、调试程序才是至关重要的。通过实际的编程和积极的思考,读者可以很快掌握很多的编程技术,而且在编程中会积累许多宝贵的编程经验。在当前的软件开发环境下,这种编程经验对开发者来说是不可或缺的。

本书的学习资源可以在清华大学出版社网站(<http://www.tup.tsinghua.edu.cn/>)按作者名检索下载。为了便于读者学习,还提供了教学视频(百度云盘 http://pan.baidu.com/pcloud/album/info?uk=1057890276&album_id=3399665821087204687)对每个游戏实例进行详细讲解。

本书由夏敏捷(中原工学院)主持编写,张秋闻(郑州轻工业学院)编写第 1 章,张锦歌(河南工业大学)编写第 2~5 章,赵一丁(中原工学院)编写第 7~9 章,樊银亭(中原工学院)和张西广(中原工学院)编写第 10~15 章,其余章节由夏敏捷编写。本书作者长期从事 Visual Basic 和 Visual Basic.NET 教学与应用开发,在长期的工作中积累了丰富的经验和教训,了解学生学习编程时需要什么样的书才能提高 Visual Basic.NET 开发能力,以更少的时间投入获得更强的实际应用能力。

由于时间仓促，加之作者水平有限，书中错误和疏漏之处在所难免。在此，诚恳地期望得到各领域专家和广大读者的批评指正。有任何问题，请发送邮件到 xmj@zut.edu.cn，与作者联系；也可发送邮件到 workemail6@163.com，与本书策划编辑联系。

编 者

夏敏捷

2016年6月

目 录

第 1 章 Visual Basic.NET 编程基础	1
1.1 .NET 框架概述	1
1.1.1 公共语言运行时	2
1.1.2 基础类库	2
1.1.3 Visual Studio 集成开发环境	2
1.1.4 编写两种应用程序	4
1.2 数据类型和表达式	7
1.2.1 数据类型	7
1.2.2 常量	8
1.2.3 变量	9
1.2.4 运算符	10
1.2.5 表达式	12
1.3 条件语句	13
1.3.1 单行结构条件语句	13
1.3.2 块结构条件语句	13
1.3.3 多分支结构	14
1.4 循环语句	15
1.4.1 For...Next 循环	15
1.4.2 While 循环	16
1.4.3 Do 循环	16
1.5 数组	18
1.5.1 声明数组	18
1.5.2 调整数组大小	19
1.5.3 ArrayList 类	19
1.6 过程	21
1.6.1 Sub 过程	21
1.6.2 Function 过程	22
1.6.3 参数的值传递和引用传递	23
1.6.4 常用内部函数	24
1.7 类和对象	27
1.7.1 类的声明和对象的创建	28

1.7.2	向类添加字段和属性	31
1.7.3	向类添加方法	33
1.7.4	向类添加事件	35
1.7.5	模块与类	37
1.8	继承和多态	38
1.8.1	基本继承	38
1.8.2	跨语言继承	41
1.8.3	可视化继承	41
1.8.4	重写方法	41
1.8.5	定义接口	43
1.8.6	实现接口	43
1.8.7	多态性	45
1.9	命名空间	47
1.9.1	声明命名空间	47
1.9.2	导入命名空间	48
1.9.3	常用命名空间	48
1.10	Visual Basic.NET 常用类	49
1.10.1	Console 类	49
1.10.2	Math 类	50
1.10.3	Convert 类	50
1.10.4	String 类	51
1.10.5	与窗体应用程序相关的类	52
第 2 章	连连看游戏	53
2.1	连连看游戏介绍	53
2.2	连连看游戏设计思路	54
2.3	关键技术	67
2.4	连连看游戏设计步骤	70
第 3 章	华容道游戏	78
3.1	华容道游戏介绍	78
3.2	华容道游戏设计思路	79
3.3	华容道游戏设计步骤	80
第 4 章	黑白棋游戏	91
4.1	黑白棋游戏介绍	91
4.2	黑白棋游戏设计思路	92
4.3	关键技术	92
4.4	黑白棋游戏设计步骤	94

第5章 扫雷游戏	101
5.1 扫雷游戏介绍	101
5.2 扫雷游戏设计思路	101
5.3 关键技术	102
5.4 扫雷游戏设计步骤	103
第6章 对对碰游戏	109
6.1 对对碰游戏介绍	109
6.2 对对碰游戏设计思路	109
6.3 关键技术	111
6.4 对对碰游戏设计步骤	114
第7章 推箱子游戏	128
7.1 推箱子游戏介绍	128
7.2 推箱子游戏设计思路	129
7.3 关键技术	133
7.4 推箱子游戏设计步骤	137
第8章 坦克大战游戏	148
8.1 坦克大战游戏介绍	148
8.2 坦克大战游戏设计思路	148
8.3 坦克大战游戏设计步骤	149
第9章 人物拼图游戏	164
9.1 人物拼图游戏介绍	164
9.2 人物拼图游戏设计思路	164
9.3 关键技术	165
9.4 人物拼图游戏设计步骤	165
第10章 七巧板游戏	170
10.1 七巧板游戏介绍	170
10.2 关键技术	170
10.3 七巧板游戏设计步骤	176
第11章 百变方块游戏	184
11.1 百变方块游戏介绍	184
11.2 百变方块游戏设计思路	184
11.3 百变方块游戏设计步骤	185

第 12 章 俄罗斯方块游戏	197
12.1 俄罗斯方块游戏介绍	197
12.2 俄罗斯方块游戏设计思路	197
12.3 俄罗斯方块游戏设计步骤	198
第 13 章 贪吃蛇游戏	211
13.1 贪吃蛇游戏介绍	211
13.2 贪吃蛇游戏设计思路	211
13.3 贪吃蛇游戏设计步骤	212
第 14 章 网络版中国象棋	229
14.1 中国象棋介绍	229
14.2 网络版中国象棋设计思路	230
14.3 关键技术	232
14.4 网络版中国象棋设计步骤	240
第 15 章 飞碟跳棋游戏	256
15.1 飞碟跳棋游戏介绍	256
15.2 飞碟跳棋游戏设计思路	256
15.3 飞碟跳棋游戏设计步骤	257
第 16 章 数独游戏	265
16.1 数独游戏介绍	265
16.2 数独游戏设计思路	265
16.3 关键技术	266
16.4 数独游戏设计步骤	267
第 17 章 拱猪扑克牌游戏	275
17.1 拱猪扑克牌游戏介绍	275
17.2 拱猪扑克牌游戏设计思路	277
17.3 拱猪扑克牌游戏设计步骤	278
第 18 章 网络聊天	299
18.1 网络聊天介绍	299
18.2 网络聊天设计思路	299
18.3 关键技术	302
18.4 客户端程序设计步骤	306
18.5 服务器端程序设计步骤	312

第 19 章	智力问答游戏	319
19.1	智力问答游戏介绍	319
19.2	智力问答游戏设计思路	319
19.3	关键技术	321
19.4	智力问答游戏设计步骤	321
第 20 章	网络五子棋游戏	328
20.1	网络五子棋游戏介绍	328
20.2	网络五子棋游戏设计思路	328
20.3	关键技术	329
20.4	网络五子棋游戏设计步骤	329
第 21 章	涂鸦游戏	338
21.1	涂鸦游戏介绍	338
21.2	关键技术	338
21.3	涂鸦游戏设计步骤	342
第 22 章	21 点扑克牌游戏	354
22.1	21 点扑克牌游戏介绍	354
22.2	关键技术	354
22.3	21 点扑克牌游戏设计步骤	355

Visual Basic.NET 语言是从 Visual Basic 演变而来的。它是现代、简单、完全面向对象和类型安全的。作为一名 Visual Basic 程序员，学习 Visual Basic.NET 其实是比较简单的，需要掌握的新知识并不多。本章重点讲解 Visual Basic.NET 和 Visual Basic 有区别的地方，相同的部分将一带而过，这样的对比学习可以让读者在已有知识的前提下快速掌握 Visual Basic.NET。

1.1 .NET 框架概述

Visual Basic.NET 提供了很多新的功能、概念和观点，学习 Visual Basic.NET 不仅要掌握语法，还需要理解并运用这些新的功能、概念和观点。当然也需要掌握 .NET 框架，理解 CLR（Common Language Runtime，公共语言运行时）。

.NET 框架（Framework）是一组用于帮助开发应用程序的类库集。Visual Studio 开发平台需要此类库集的支持，用 Visual Studio 开发的程序也需要此类库集的支持。

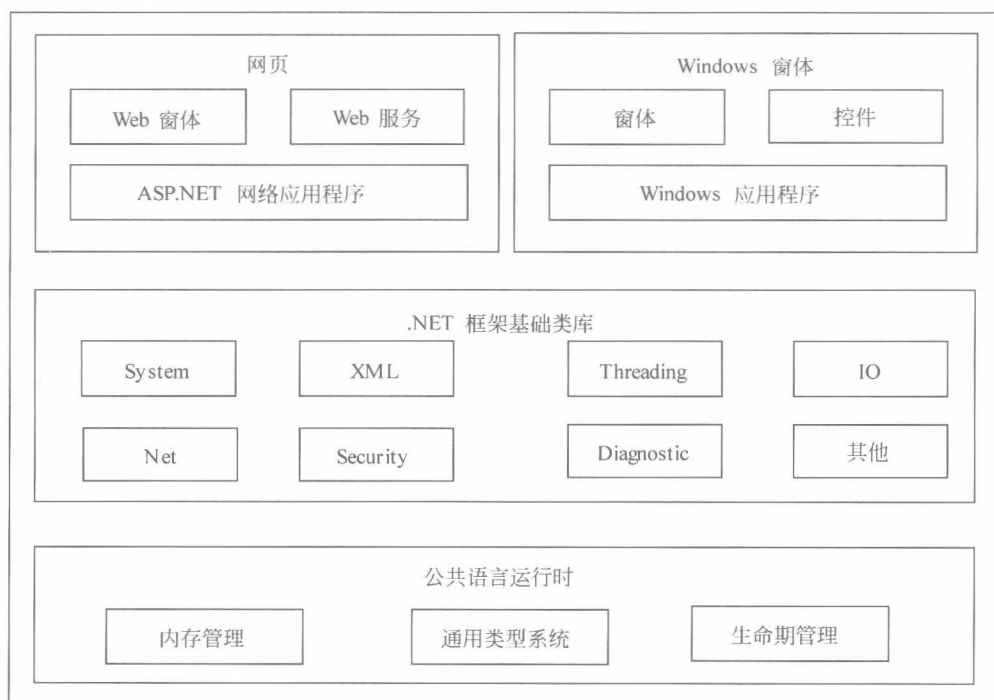


图 1-1 .NET 框架的整体结构

1.1.1 公共语言运行时

1. 公共语言运行时概述

公共语言运行时是 .NET 框架的基础, 提供诸如内存管理、线程管理和远程处理等核心服务, 强制实施严格的类型安全以及确保安全性和可靠性的其他形式的代码的准确性。事实上, 代码管理的概念是公共语言运行时的基本原则。以运行时为目标的代码称为托管代码, 而不以运行时为目标的代码称为非托管代码。

2. 即时编译

可以直接将源程序编译为 EXE 或者 DLL 文件, 但是编译出来的程序代码并不是 CPU 能直接执行的机器代码, 而是一种中间语言 (Intermediate Language, IL) 代码。在 IL 代码被调用执行时, CLR 的 Class Loader 会将需要的 IL 代码装入内存, 然后通过即时 (Just-In-Time) 编译方式将其临时编译成所用平台的 CPU 可直接执行的机器代码 (二进制文件)。人们看到的程序运行, 其实是经过即时编译后的二进制文件在执行。这时可能有很多朋友会问, 这种二次编译是否多此一举, 具体实现中是否很麻烦, 是否会影响程序的运行速度? 二次编译的确会影响程序运行的速度, 但它为实现跨平台带来了可能。其实这种编译过程犹如 Java 中的 JVM (Java 虚拟机)。正是因为有了 JVM, 才使 Java 能够开发出跨平台的应用程序。二次编译是 CLR 在 .NET 框架下实现 IL 文件到二进制文件的转换, 它是自动完成的, 并不需要人员的参与, 所以它不会给程序执行带来麻烦。

1.1.2 基础类库

类库 (Class Library) 是一个由 .NET Framework SDK (Software Development Kit, 软件开发工具包) 中包含的类、接口和值类型组成的库。该库提供对系统功能的访问, 是建立 .NET Framework 应用程序、组件和控件的基础。

类库提供了 Internet 和企业级开发所需要的各种功能, 完全支持 Web 标准及其应用, 而且使用简单, 扩充方便。类库被分配在不同的命名空间下, System 命名空间是 .NET Framework 中基本类型的根命名空间。

1.1.3 Visual Studio 集成开发环境

Visual Studio 集成开发环境包含 Visual Basic.NET、Visual C++.NET、Visual C# .NET 和 Visual J#.NET 等 4 种编程语言, 并允许它们共享工具, 这有助于创建混合语言解决方案。

Visual Studio 集成开发环境 (见图 1-2) 与以前的 VB 6.0 集成开发环境类似, 但提供了更方便、更可靠的方式来开发应用程序。Visual Studio 集成开发环境是一个集界面设计、代码编写、程序调试和资源管理于一体的工作环境。用户可以依靠开发环境提供的控件、窗体和方法等进行各种应用程序的开发, 减少了代码编写工作量, 更注重程序逻辑结构的设计, 极大地提高了程序开发效率。本小节将对 Visual Studio 2010 开发环境进行简单介绍。

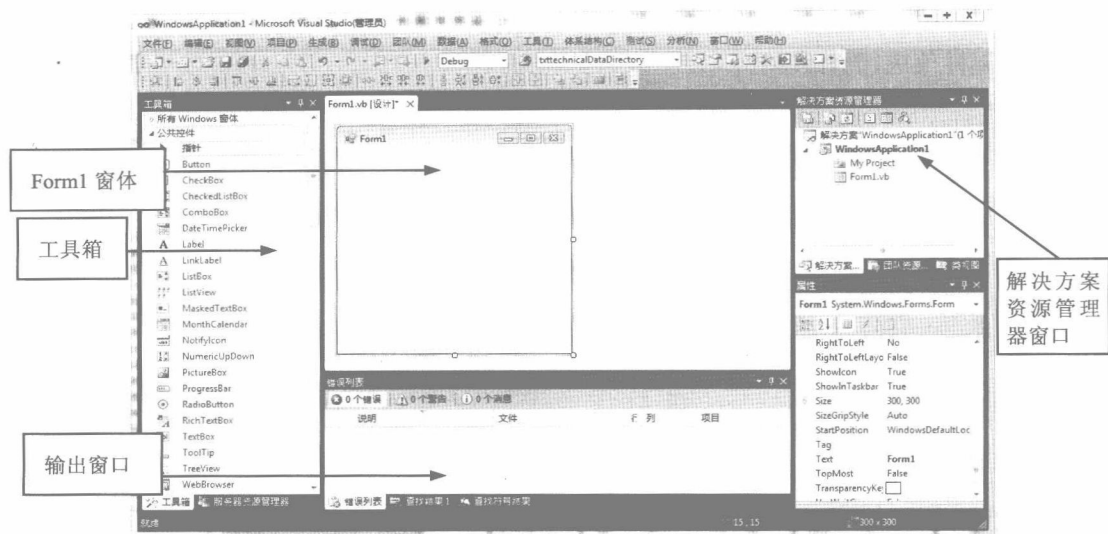


图 1-2 Visual Studio 2010 集成开发环境

Visual Studio 开发环境主要包括菜单栏、工具栏、窗体设计器、工具箱、【属性】窗口、解决方案资源管理器和代码编辑器等。

1. 解决方案资源管理器 (Solution Explorer)

使用 Visual Studio 开发的应用程序都称为解决方案，每一个解决方案可以包含一个或多个项目。一个项目通常是一个完整的程序模块，可以有多个项。【解决方案资源管理器】窗口显示 Visual Studio 解决方案的树型结构。在解决方案资源管理器中可以浏览组成解决方案的所有项目和每个项目中的文件，可以对解决方案的各元素进行组织和编辑。

一个项目中通常有非常多的文件，但是其中的大部分文件，如 My Project 组中的文件，不需要开发者进行直接的编辑。开发者所要做的是双击解决方案资源管理器中的 My Project 图标，打开属性编辑器对项目进行配置。

2. 工具箱

工具箱用于向 Windows 应用程序或 Web 应用程序添加控件。工具箱使用选项卡的方式分类管理其中的控件。根据当前正在使用的设计器或编辑器，工具箱中可用的选项卡和控件会有所变化。在 Visual Studio 开发环境左边的工具箱中列出了可以应用到窗体上的控件，如图 1-3 所示。

3. 窗体设计器

在开发环境的中部是软件界面的主窗口，用来显示指定的窗体。窗体是一小块屏幕区域，通常为矩形，可用来向用户显示信息并接收用户的输入。设计窗体的用户界面时，最简单的方法是将控件放在其表面上。默认情况下，窗体处于设计视图状态下。在解决方案资源管理器中单击“查看代码”按钮后，会切换到程序代码编辑窗口；单击“视图设计器”按钮后，会切换到窗体设计视图。

4. 属性 (Properties)

【属性】窗口如图 1-4 所示，默认位于 Visual Studio 主界面的右下角。与开发环境的许多其他窗口一样，如果关闭【属性】窗口，还可以单击工具栏中的按钮，重新打开该窗

口。窗体和控件都有自己的属性,【属性】窗口列出了窗体或控件的属性,可以通过它对窗体控件的属性值进行修改。

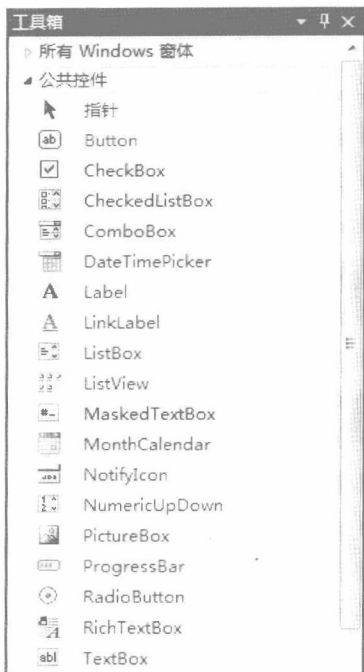


图 1-3 工具箱



图 1-4 【属性】窗口

例如,在窗体设计视图选择窗体 Form1,则【属性】窗口就会显示 Form1 的属性,如图 1-4 所示。在其中找到 Text 属性,把默认的 Form1 改为“第一个窗体”。一旦接受属性的改变,新值就会显示为窗体的标题。与在其他开发环境中通过用户界面编辑的属性隐藏在项目的一些二进制文件或专用部分不同,.NET 属性是在源文件中定义的。因此,尽管【属性】窗口看起来类似于其他开发环境,如 VB 6.0,但在 Visual Studio 中它要强大得多。

5. 输出

【输出】窗口用来显示程序运行时产生的信息,包括开发环境给出的信息,如编译项目时产生的错误以及在程序中设定的要输出的信息等。

提示:本书所用的 Visual Studio 的版本是 2010。

1.1.4 编写两种应用程序

在 Visual Basic.NET 中可编写两种应用程序。一种是控制台应用程序,也称为 Console 应用程序,用于在命令行方式下运行。另一种是 Windows 应用程序,运行界面是 Windows 窗口。

下面是在 Visual Basic.NET 中编写控制台应用程序的基本步骤。

(1) 启动 Visual Studio,选择【文件】→【新建项目】命令后,弹出如图 1-5 所示的【新建项目】对话框。

(2) 将项目类型设置为 Visual Basic，模板设置为【控制台应用程序】。

(3) 在【名称】文本框中输入“ch1_01”，然后单击【确定】按钮。

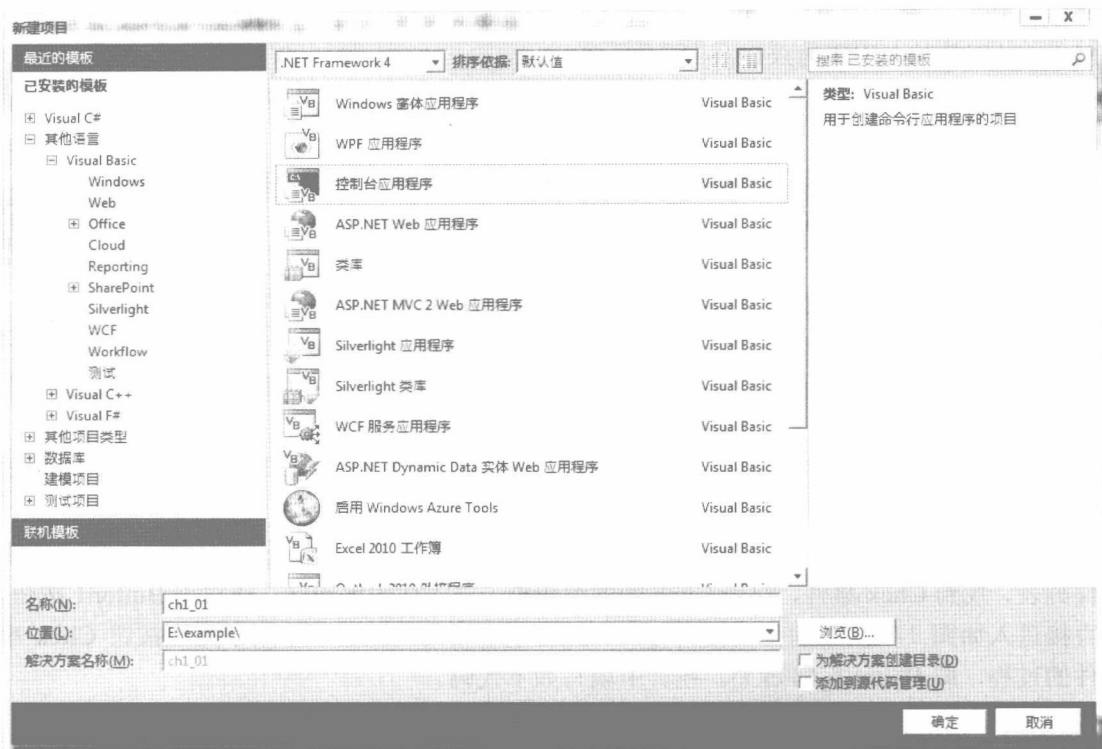


图 1-5 【新建项目】对话框

(4) 将自动生成的 Module1.vb 文件更改为下面的内容。

```
Module Module1
    Sub Main()
        Console.WriteLine("Hello World")
        Console.Read()
    End Sub
End Module
```

Console.WriteLine("Hello World!")中，Console 是 System 命名空间的一个类，它有一个 WriteLine 方法，其作用和 Visual Basic 的 print 一样，输出一行字符串。

(5) 单击【启动调试】按钮或按 F5 键，Visual Studio 会自动完成编译工作，并会弹出窗口显示“Hello World”。

(6) 选择【文件】→【全部保存】命令后，弹出【保存项目】对话框。在该对话框中通过【浏览】按钮选择项目的保存位置“E:\example”，取消选中【创建解决方案的目录】复选框，后单击【保存】按钮。这样在“E:\example”文件夹中就产生了名为 ch1_01 的文件夹，并在里面创建了 ch1_01.vbproj 项目文件（包含在“ch1_01\ch1_01”文件夹下）、Module1.vb 程序文件（包含在“ch1_01\ch1_01”文件夹下）和 ch1_01.sln 解决方案文件等。

在 Visual Basic.NET 中,所有的过程或函数都必须封装在类或模块中。一个类可以有成员变量和方法(过程或函数)。然而,模块只包含过程或函数,并且不能进行实例化。就 ch1_01 应用程序来说,Module1 模块包含一个 Main()过程。在 Visual Basic.NET 程序中,Main()过程是整个程序的入口。

在应用程序加载到内存之后,Main()过程就会接收控制,因此应该将应用程序的启动代码放在此过程中。

下面介绍如何实现显示“Hello World”的 Windows 应用程序,基本步骤如下。

(1) 启动 Visual Studio,选择【文件】→【新建项目】命令后,弹出【新建项目】对话框。

(2) 将项目类型设置为 Visual Basic,模板设置为【Windows 窗体应用程序】。

(3) 在【名称】文本框中输入“ch1_02”,然后单击【确定】按钮。

(4) 在【Form1.vb[设计]】窗口(见图 1-2)中,从工具箱的【所有 Windows 窗体】选项卡中往 Form1 窗体中拖入 Button 控件,产生 Button1 按钮。

(5) 在【属性】窗口中设置 Button1 按钮的 Text 属性为“欢迎”;设置 Form1 窗体的 Text 属性为“第一个窗体”。

(6) 在【属性】窗口中选择命令按钮,单击【事件】按钮,出现如图 1-6 所示的事件列表,找到 Click 事件,双击 Click 旁的编辑框,进入代码编辑区;或双击 Button1 按钮,直接进入如图 1-7 所示的代码编辑窗口。此时系统自动生成了一个用于处理按钮 Click 事件的过程,即 Button1_Click()。在其中填写如下代码。

```
MessageBox.Show("Hello world")
```

(7) 单击【启动调试】按钮或按 F5 键,Visual Studio 会自动完成编译工作,并生成 Windows 应用程序。

(8) 选择【文件】→【全部保存】命令,弹出【保存项目】对话框。在该对话框的【位置】文本框中输入文件夹名“E:\example”,然后单击【保存】按钮。这样在“E:\example”目录中就产生了名为“ch1_02”的文件夹,并在其中创建了名为“ch1_02.vbproj”的项目文件。



图 1-6 显示事件的【属性】窗口



图 1-7 代码编辑窗口

对于其他的 Windows 程序，其编写的基本步骤是相似的，只不过程序中涉及更多的控件、类和其他的调用。

在解决方案资源管理器中可以看到 Form1.vb 窗体模块文件，其实它是保存窗体中各控件的属性内容及程序代码的文件。单击【查看代码】按钮，可查看 Form1.vb 程序源代码。

```
Public Class Form1
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
    MessageBox.Show("Hello world")
End Sub
End Class
```

注意：MessageBox.Show() 用于弹出消息框。

1.2 数据类型和表达式

1.2.1 数据类型

为了方便识别和处理，编程语言系统中的不同信息在计算机中具有不同的表示，占用不同的存储空间，这些信息在语言系统中称为数据类型。

Visual Basic.NET 中的数据类型和 Visual Basic 是类似的。Visual Basic.NET 使用的每种数据类型都与公共语言运行时（CLR）的通用类型系统（Common Type System，CTS）中的一种类型相对应。这为构建跨语言集成，保证代码类型安全，确保用不同语言编写的对象能够交互作用打下了基础。

Visual Basic.NET 支持的数据类型包括：所有数值类型的数据类型（见表 1-1）、String、Char、Boolean、Date、结构、枚举类型、类等其他类型（见表 1-2）。

表 1-1 数值类型的数据类型

数据类型	关键字	类型符	说明
整型	Integer	%	用于表示简单整数，取值范围为-2 147 483 648~2 147 483 647
字节型	Byte		用于简单算术运算。由于这个类型的变量可以在一个字节中存储，所以运算速度最快。其取值范围为 0~255
短整型	Short		是整型的一种形式，相对表示范围较小。其取值范围-32 768~32 767
长整型	Long	&	是整型的一种形式，相对表示范围较大
单精度型	Single	!	用于存放单精度浮点数
双精度型	Double	#	用于存放双精度浮点数
小数	Decimal	@	常用于存储货币值

表 1-2 其他类型

数据类型	关键字	类型符	说明
字符串型	String	\$	用于存放任何形式的字符串，包括一个字符或多行字符
字符型	Char	C	用于存放一个字符，它以 0~65 535 之间的数值的形式存储