第1章 认识 VB.NET

Visual Basic.NET 语言是从 Visual Basic 语言演变而来的, 是一种比较流行的、简单易学的、 功能强大的应用程序开发工具。本章简单介绍了有关.NET 的一些基本概念、VB.NET 集成开发 环境以及如何使用 VB.NET 进行简单的程序设计。

1.1 VB.NET 概述

1.1.1 Visual Basic 的发展

BASIC 于 1964 年诞生,其含义为"初学者通用的符号指令代码",由于它简单易学而一直被大多数初学者作为入门首选的程序设计语言。1976 年前后开发出 DOS 环境下的 GW-BASIC,20世纪 80 年代中期又出现了多种结构化 BASIC 语言,如 Quick BASIC、True BASIC 等。

1988 年微软公司推出 Windows 操作系统,从此进入了鼠标操作的图形用户界面时代,同时 开发在 Windows 环境下的应用程序成为 20 世纪 90 年代软件开发的主导潮流。起初人们在开发 Windows 应用程序时遇到了很大困难,因为要编写 Windows 环境下运行的程序,必须建立相应 的窗口、菜单、对话框等各种"控件",程序的编写变得越来越复杂。

1991 年微软公司推出 VB1.0,使这种情况有了根本的改观。VB 除提供常规的编程机制外,还提供了一套可视化的编程工具,非常适合编程人员创建图形用户界面。VB 以可视化工具为界面设计,以结构化 BASIC 为基础,以事件驱动为运行机制,它的诞生标志着软件设计和开发进入了一个新时代。

随着 Windows 操作平台的不断成熟, VB 经历了从 VB 1.0 至 VB 6.0(1998 年)的多次版本 升级,其功能逐步增强,应用范围越来越广。

2002 年微软公司推出 VB.NET 7.0,随后又陆续发布多种 VB.NET 版本。VB.NET 是微软公司改进 VB 语言的新一代产品。

1.1.2 什么是.NET

随着 Internet 的不断发展和广泛应用, Internet 逐渐成为编程领域的中心,为适应这种新局面的变化,2000年6月微软公司提出了其新一代基于 Internet 平台的软件开发构想——.NET 战略,并推出了.NET 开发平台。

如同 MS-DOS 和 Windows 一样,..NET 将大大改变我们的计算领域。它以 Internet 为基础, 采用 Internet 上标准的通信协议,允许应用程序通过 Internet 进行通信和共享数据,而不管所采用的是哪种操作系统、设备或编程语言。

.NET 开发平台包括.NET 框架(.NET Framework)、.NET 开发技术和.NET 开发工具等组成 部分。其中.NET 框架是一个集成在 Windows 系统中的组件,是构建以 Internet 为开发平台的基础工具,它包括公共语言运行时库(CLR)和基础类库。基础类库提供了大量可重用的类,无论 是 VB.NET,还是 VC++.NET,都使用同一类库来开发软件。

1.1.3 VS.NET 与 VB.NET

Visual Studio.NET(简称 VS.NET)是微软公司推出的第一个基于.NET 框架的应用程序开发 工具,它把 VB.NET、VC++.NET、VC#.NET 等集于一体,提供了可视化的、高效的、多编程语 言的,可以创建、测试和组织应用程序的集成开发环境。

VB.NET 是 VS.NET 支持的多种编程语言之一,也是 VS.NET 中最早推出的应用程序开发工 具。VB.NET 是 VB 的全新版本,它继承了传统 Visual Basic 的特点和风格,但又不是 VB 6.0 的 简单升级版。它从功能上消除了许多 VB 原有的局限性,如面向对象的能力较弱,很难满足大型 项目的开发需求等,新增和加强了许多面向对象特性,体现了真正的面向对象的程序设计思想, 使其与 C++和 Java 这类高级语言一样,成为功能强大的应用程序开发工具。VB.NET 不仅可以快 速开发 Windows 应用程序,并且可以非常容易地开发适用于 Internet 的 Web 程序。

本书以 VB.NET 2010 版为背景,相应的.NET 框架为.NET Framework 4.0。

1.1.4 VB.NET 的主要特点

① 面向对象的可视化设计。VB.NET 采用了面向对象的程序设计方法(OOP),把程序和数据"封装"起来作为一个对象。所谓"对象"就是一个可操作的实体,如窗体、命令按钮、文本框、标签等。程序设计时编程人员不必为界面设计编写大量程序代码,只需利用系统提供的工具,直接在窗体上建立各种控件对象,并为每个控件对象设置属性。

② 事件驱动的编程机制。VB.NET 通过事件来执行对象的操作,事件可由用户的操作触发, 也可以由系统或应用程序触发。例如,命令按钮是一个对象,当用户单击该按钮时,将触发一个 "单击"(Click)事件,而在该事件发生时,系统将自动执行相应的事件过程,用以实现指定的操 作和达到运算、处理的目的。

在 VB.NET 中,编程人员只需针对这些事件编写相应的处理代码,这样的代码一般较短,所 以程序既易于编写又易于维护。

③ 软件的集成式开发。VB.NET 集成在 VS.NET 中,用户可以充分利用所有.NET 平台特性,使用 VB.NET 集成开发环境方便地设计界面、编写代码、调试程序和保存文件。

VS.NET 中所有语言使用统一的开发环境,因此 VB.NET 与其他语言之间的数据和代码交换更加方便,极大地简化了应用程序的开发,提高编程效率。

④ 支持结构化程序设计。VB.NET 是在结构化的 BASIC 语言基础上发展起来的,加上面向 对象的设计方法,因此是更出色的结构化程序设计语言。

⑤ 强大的数据库功能。VB.NET 采用 ADO.NET 数据访问技术,对多种不同类型的数据库 (如 Oracle、Access、SQL Server 等)中的数据,以统一的方式管理和访问。

⑥ 网络功能。VB.NET 提供了更直观、方便的 Web 应用程序开发环境,可以通过 Web Server 实现跨平台的功能调用和使用 XML 来进行数据交换,能有效地建立全交互的互联网网站。

1.1.5 VB.NET 的启动与退出

VS.NET 作为一个集成的开发环境,将以往独立的诸如 VB、VC++、ASP 等开发工具都整合 在了.NET 框架开发平台中。安装 VB.NET 的过程其实就是安装整个 VS.NET 的过程,也只有安 装了所有的开发语言和控件,才可以得到一个可以跨越多种语言的开发环境。

1. 启动 VB.NET

VB.NET 是 VS.NET 的一部分,因此启动 VB.NET,实质上是启动 VS.NET。具体方法是:

从"开始"菜单中选择"所有程序",指向"Microsoft Visual Studio 2010",单击级联菜单中的 "Microsoft Visual Studio 2010",即可启动 VS.NET,进入"起始页",如图 1.1 所示。

	∞ 起始页 - Microsoft Visual Studio(管理员)		6 ×
	文件(E) 瞬期(E) 初期(A) 測试(D) 間(A(M) 教研(A) 工具(D) 到(式S) 銀口(M) 琴助(H)		
	[]] 4 [] 그·원 4 [1] 2 [] 3 [] 2 [] 2 · 6 · 년·다 (1) 그 그 전 년 1] 6 [] 4 [] 3 [] 3	父童心口。	
起始页 -	Robert ×		
新建项目 🔪	↓ 正規則 Team Foundation Server 安禄使用 Windraws Web 〒 Office Sha	arePoint 教授	
打开项目 -	The second secon	95%所被75% 第0%用。	
最近使用	■近使用的項目	33 27 - 27 - 27 - 27 - 27 - 27 - 27 - 27 -	
的项目			
	書 一 File Visual Studio		
	社区和学习资源		
\langle	☑ 在波用加載所共和此页 図 和設計型形式四		
	Dive		

图 1.1 VS.NET 的起始页

说明:在"起始页"底部有两个复选框(见图 1.1 底部的椭圆线条部分),可以设置启动时 是否显示"起始页",项目加载后是否关闭"起始页"。

2. 新建项目

在 VB.NET 中,编写 VB.NET 应用程序意味着创建一个项目。一个项目由存放在独立文件夹中的若干文件组成。新建一个 VB.NET 项目,有以下两种常用方法。

方法 1: 启动 VB.NET 后,在起始页上单击"新建项目"按钮。

方法 2: 执行主窗口的"文件"菜单中的"新建项目"命令。

用以上方法均可以打开"新建项目"对话框,如图 1.2 所示。"新建项目"对话框提供了一 组与所要创建的应用程序类型相关的模板选项。

新建项目					? 🗙
最近的模板		.NET Fr	amework 4 🔹 排序依据: 默认值	• 11 11	搜索已安装的模板 👂
已安装的模板		Vo	14. ⁴ J	Marcal Barda	类型: Visual Basic
✓ Visual Basic			Windows 图体应用程序	Visual Basic	用于创建具有 Windows 用户界面的应用
Windows Web		(10)	WPF 应用程序	Visual Basic	程序的项目
Cloud		Ē.	控制台应用程序	Visual Basic	
Reporting SharePoint		W ^V B	类库	Visual Basic	
Silverlight WCF		VB	WPF 浏览器应用程序	Visual Basic	
Workflow 测试		VB T	Windows 服务	Visual Basic	
▷ 其他语言 ▷ 其他项目类型		₹ ^V B	WPF 用户控件库	Visual Basic	
▷ 数据库 ▷ 測试项目		₩_V _B	WPF 自定义控件库	Visual Basic	
联机模板		VB	空項目	Visual Basic	
		VB #	Windows 窗体控件库	Visual Basic	
SP PT (NI).	WindowsApplie	cation1			
1000(1 <u>8</u>):	windowsApplie	Lauoni			
					确定 取消

图 1.2 "新建项目"对话框

在"新建项目"对话框的.NET 框架类型下拉列表中默认选择".NET Framework 4",在左侧"已 安装的模板"框中,展开"Visual Basic"分支,选择"Windows"项,在中间选择"Windows 窗

体应用程序",在下方的"名称"框中输入要创建的项目名称,或采用系统给定的默认名,单击"确定"按钮,即可创建一个新项目,进入 VB.NET 集成开发环境,如图 1.3 所示。



图 1.3 VB.NET 集成开发环境

3. 退出 VB.NET

如果要退出 VB.NET,可单击 VB.NET 主窗口右上角的"关闭"按钮,或选择"文件"菜单中的"退出"命令,VB.NET 会自动判断用户是否修改了项目的内容,询问用户是否保存文件或 直接退出。

1.2 VB.NET 集成开发环境

VB.NET 集成开发环境由许多窗口组成,根据不同的应用程序开发需要使用不同的窗口,以下介绍最常用的窗口。

1.2.1 主窗口

主窗口主要由标题栏、菜单栏和工具栏等组成。

1. 标题栏

标题栏主要用于显示应用程序的名称及其工作状态。新建 VB.NET 项目后,标题栏中显示的 信息为:

项目名称 - Microsoft Visual Studio

其中"项目名称"为建立项目时由用户给定(系统默认的项目名称为 WindowsApplicationX, X 为 1, 2, …),并表明当前的工作状态处于"设计模式"。随着工作状态的不同,标题栏上的显示信息也随之改变。

VB.NET 有三种工作模式:设计模式、运行模式、调试模式。

设计模式:可以进行用户界面的设计和代码的编写。

运行模式:应用程序运行阶段,在标题栏中显示"项目名称(正在运行)-Microsoft Visual

• 4 •

Studio",此时不可以编辑代码,也不可以编辑界面。

调试模式:应用程序运行暂时中断,在标题栏中显示"项目名称(正在调试)-Microsoft Visual Studio",在此模式下可以进行程序的调试。此时可以编辑代码,但不可以编辑界面。

2. 菜单栏

通常,菜单栏中包括 13 项下拉菜单,提供了用于开发、调试和保存应用程序所需的所有 命令。

3. 工具栏

工具栏可以迅速地访问常用的菜单命令。VB.NET 提供了标准工具栏、布局工具栏、调试工 具栏、格式设置工具栏等 30 多个专用的工具栏。要显示或隐藏工具栏,可以选择"视图"菜单 中的"工具栏"命令,或右击标准工具栏,在快捷菜单中选中或取消选中所需的工具栏。

在一般情况下,集成开发环境中只显示"标准"工具栏(简称工具栏)。

1.2.2 工具箱窗口

工具箱窗口(简称工具箱)如图 1.4 所示,它提供了建立应用程序的各种控件。工具箱位于

集成开发环境的左侧,默认情况下是自动隐藏的,当鼠标接近工具箱 敏感区域时,工具箱会自动弹开,当鼠标离开时又会自动隐藏。如果 关闭了工具箱窗口,可以执行"视图"中的"工具箱"命令,或单击 工具栏中的"工具箱"按钮,将其打开。

说明:①单击工具箱标题栏上的"自动隐藏"按钮,可取消自动 隐藏功能,使工具箱始终保持为打开状态;②为方便使用,通常通过 工具箱标题栏上的"窗口位置"下拉列表,将工具箱设置为"停靠" 状态。

VB.NET 将控件分类放置在工具箱的不同选项卡中,常用的选项 卡有"所有 Windows 窗体"、"公共控件"、"容器"、"菜单和工具"、"对 话框"等。例如,在"所有 Windows 窗体"选项卡中,放置了常用的 文本框、标签框、命令按钮等控件。

以下简要介绍控件的一些基本操作方法。

1. 在窗体上添加控件

常用以下两种方法。

 单击工具箱中所需的控件按钮,在窗体上按住鼠标左键拖动, 则可添加控件。

② 双击工具箱中所需的控件按钮,即可在窗体左上角创建一个控件,然后移动控件到合适位置。

T县箱 - 4 × ▷ 所有 Windows 窗体 ┛ 公共控件 ▶ 指针 Button CheckBox CheckedListBox ComboBox DateTimePicker A Label A LinkLabel ListBox ListView #_ MaskedTextBox MonthCalendar NotifyIcon NumericUpDown 1 * PictureBox (III) ProgressBar (\bullet) RadioButton 8 4 RichTextBox abl TextBox 🖢 ToolTip TreeView WebBrowser ▲ 容器 ▶ 指针 FlowLayoutPanel GroupBox

图 1.4 工具箱窗口

2. 控件的缩放、移动、复制和删除

在设计阶段,选定(单击)窗体上的某个控件时,控件的边框上会出现控点(见图 1.10),

这表明该控件处于"活动"状态,或称为"当前控件"。

说明:不同控件被选定后出现的控点数有所不同,有些是8个,有些是1或2个。

 缩放:选定控件后,把鼠标指针指向某一控点,当出现双向箭头时,按住鼠标左键拖动, 可以改变控件的大小。

② 移动:选定控件后,把鼠标指针指向控件的内部,当出现十字箭头时,按住鼠标左键拖动,即可移动控件的位置。

③ 复制:选定控件后,单击工具栏上的"复制"按钮,再单击"粘贴"按钮,即可添加一 个与选定控件同类的控件。

④ 删除:选定控件后,按 Delete 键或选择"编辑"菜单中的"删除"命令。

3. 选定多个控件

要调整多个控件,需要先同时选定多个控件,常用方法有两种。

① 在窗体的空白区域中按住鼠标左键拖动拉出一个矩形框,框住需要选定的多个控件。

② 在按 Shift 键的同时,用鼠标依次单击所要选定的控件。

1.2.3 解决方案资源管理器窗口

为便于管理,VB.NET 中引入了两类容器:解决方案资源管理器和项目。解决方案包含了开 发一个应用程序的所有组成部分(如文件夹、文件、引用、数据连接等)。一个解决方案可以由 一个或多个项目组成。项目是一个独立的编程单位,可以用不同的语言开发。每个项目包含有窗 体文件和其他一些相关的文件。

说明:本书所有解决方案都只包含一个项目,即一个应用程序只有一个项目;一个项目中可 以建立一到多个窗体,本书前几章介绍的项目中只包含单个窗体,从第8章开始才引入多窗体的 概念。

解决方案资源管理器如图 1.5 所示,其作用是查看和管理解决方案中的项目。解决方案资源 管理器是 VB.NET 的文件管理器,其功能类似 Windows 资源管理器,它以树状的结构显示整个 解决方案中包括哪些项目以及每个项目的组成信息,也可以对项目中的文件(如窗体文件)进行 复制、删除、重命名等操作。



图 1.5 解决方案资源管理器窗口

如果关闭了解决方案资源管理器窗口,可以选择"视图"菜单中的"解决方案资源管理器" 命令来打开该窗口。

解决方案资源管理器窗口的工具栏上常用按钮及其作用如下。

① "属性" 按钮: 打开"属性" 窗口,显示所选定对象的属性。

②"显示所有文件"按钮: 解决方案资源管理器隐藏了一些文件,单击该按钮可以显示出这些隐藏的文件。

③"查看代码"按钮:切换至代码窗口。

④"视图设计器"按钮:切换至窗体设计器窗口。

1.2.4 窗体设计器窗口

应用程序的窗口在设计阶段称为"窗体",每个窗体都有自己的窗体设计器。窗体设计器窗 口(简称窗体窗口)如图 1.6 所示,它是设计应用程序用户界面的场所。



图 1.6 窗体设计器窗口

当创建一个新项目时, VB.NET 会同时创建一个新的窗体,其默认名为 Form1,一个应用程 序可以有多个窗体,可通过"项目"菜单中的"添加 Windows 窗体"命令来添加新窗体。在窗体 中,用户可以根据需要添加相应的控件,并设置相应的属性来创建应用程序的界面。

1.2.5 代码设计窗口

代码设计窗口(简称代码窗口)如图 1.7 所示,用来显示和编辑程序代码。从"视图"菜单 中选择"代码"命令,或者用鼠标双击窗体或窗体中的一个控件,或者单击"解决方案资源管理 器"窗口的"查看代码"按钮等,可以打开代码窗口。

对象列表框 	代码纲	a辑区 	过程3	刘表框
Form1.vb × Form1.vb [设计]				.
⁴ \$Form1		▼ 🎒 (声明)		•
⊟Public Class Form1				‡ ^
End Class				

图 1.7 代码设计窗口

代码窗口有如下主要内容。

① 对象列表框:列出当前窗体及其所包含的所有对象名。

② 过程列表框:列出所选定对象的所有事件过程名和用户自定义过程名。

编写事件过程时,在对象列表框中选择对象名,在过程列表框中选择事件名,即可在代码编 辑区中形成对象的事件过程模板,用户可在该模板内输入和编辑代码。

③ 代码编辑区:编辑程序代码的区域。当新建一个新窗体时,系统会自动提供"Public Class FormX"及"End Class"的模板,用于声明名称为FormX(X为1、2、…)的新类。

默认情况下, VB.NET 以大纲方式显示源代码, 用户可以通过单击代码行左侧的"+" 或"-"来展开或折叠代码。选择"编辑"菜单中的"大纲显示"命令, 可以设定代码的显 示方式。

为了便于代码的编辑,代码编辑器提供了"智能感知选项"、"自动语法检测"、"自动缩进" 等功能。

1.2.6 属性窗口

属性窗口如图 1.8 所示,它主要用于显示和更改所选定对象的属性。每个对象都由一组属性 来描述其特征,如颜色、字体、大小等,在程序设计时,可以通过属性窗口来设置或修改对象的 属性。



图 1.8 属性窗口

属性窗口由以下部分组成。

 ① 对象列表框:显示当前所选定对象的名称及所属的类。单击其下拉按钮,可列出项目中 全部对象的名称,从中可以选择要设置属性的对象。

② 选项按钮:常用的左边两个选项按钮,分别是"按分类顺序"和"按字母顺序",可选择 其中一种排列方式,显示当前选定对象的属性。

③ 属性列表框:属性列表框由中间一条直线将其分为两部分,左边列出的是当前选定对象的属性名称,右边列出的是对应的属性值,可对该属性值进行设置或修改。如果属性值右侧有"…" 或"▼"按钮,表示有预定值可供选择,

1.2.7 其他窗口

除上述几种常用窗口外,在集成开发环境中还有其他一些窗口,如即时窗口、输出窗口、命 令窗口、对象浏览器窗口、任务列表窗口等,这里就不一一介绍。

1.2.8 窗口布局

在 VB.NET 集成开发环境中,窗口按照布局方式可以分为两类,一类是位置相对固定的窗口,如主窗口、窗体窗口等;另一类是可浮动、可停靠、可隐藏的窗口,如工具箱、属性窗口、解决方案资源管理器窗口等,用鼠标在可浮动窗口标题栏上右击,从快捷菜单中可选择所需操作。

执行主窗口"窗口"菜单中的"重置窗口布局"命令,可以将集成开发环境中的窗口布局恢 复到软件安装时的初始状态。

说明:初学者使用 VB.NET 时,由于操作不熟练,容易造成操作窗口布局混乱,此时可通过"重置窗口布局"命令将其重置。

1.2.9 使用帮助系统

VB.NET 的帮助系统集成在 VS.NET 的帮助系统之中,是通过 MSDN(Microsoft Developer Network) Online 提供的内容或本地安装的内容(安装 VS.NET 系统时默认安装在本地硬盘上) 来提供帮助信息的。用户可以使用 Web 浏览器联机或脱机查看这些帮助信息。

执行"帮助"→"查看帮助"命令,可以在默认浏览器中显示本地的帮助信息。

执行"帮助"→"管理帮助设置"命令,可以打开 Help Library 管理器,然后根据需要从中 选择"联机检查更新"、"联机安装内容"等操作。

在 VB.NET 集成开发环境中,常用 F1 键获取与上下文相关的帮助,具体的操作方法是:选 取需要帮助的主题,按 F1 键,则可以直接打开与之相关的帮助内容。获取帮助的主题可以是:

- 工具箱中的控件
- 窗体及窗体中的对象
- 属性窗口中的属性

● VB.NET 语言的关键字(如 For、If、End 等)

例如,若要获取对"End"语句的帮助,只要在"代码窗口"中选定该语句,按F1键,系统 就会直接显示该语句的帮助信息。

1.3 创建简单的应用程序

1.3.1 建立 VB.NET 应用程序的步骤

使用 VB.NET 编写程序,一般可分为两大部分工作:设计用户界面和编写程序代码。VB.NET 应用面向对象的程序设计方法,先要确定对象,然后才能针对这些对象进行代码设计。编写一个 VB.NET 应用程序的一般步骤如下。

- ① 创建 VB.NET 应用程序项目。
- ② 建立用户界面的对象。
- ③设置对象的属性。
- ④ 编写程序代码。
- ⑤ 保存和运行程序。

下面通过一个实例,来说明建立 VB.NET 应用程序的一般步骤和方法。

1.3.2 一个简单程序

【例 1.1】 一个简单程序实例。程序的设计界面如图 1.9 所示, 它由 1 个窗体、1 个文本框和

2 个命令按钮组成。程序运行时,文本框中初始状态为空白。单击"运行"命令按钮,文本框中 会出现"欢迎你来到 VB.NET 世界!"字样。单击"结束"命令按钮,则结束程序运行。



图 1.9 例 1.1 的设计界面

(1) 创建 VB.NET 项目

启动 VB.NET,在"起始页"窗口中单击"新建项目"按钮,或在主窗口中执行"文件"菜单中的"新建项目"命令,打开"新建项目"对话框。在对话框左侧 "已安装的模板"框中,选择"Visual Basic"分支中的"Windows"项,在中部选择"Windows 窗体应用程序"项,单击"确定"按钮,就可建立一个默认名为"WindowsApplication1"的新项目。

(2) 建立用户界面的对象

新建项目时,系统会自动提供一个空白的 Windows 新窗体,默认名为 Form1,该窗体文件 Form1.vb 也同时被添加到解决方案资源管理器窗口中。

在窗体上添加文本框 TextBox1,操作方法是:展开工具箱中的"所有 Windows 窗体"选项 卡,单击其中的 TextBox 控件,将鼠标移到窗体上适当的位置,按住鼠标左键,拖动出一个矩形 框,放开鼠标左键,就在这个矩形框中创建一个 TextBox1 控件。

使用类似上面的操作方法,可以在窗体的适当位置上添加 2 个 Button 控件,默认名称为 Button1 和 Button2。图 1.10 为添加控件后窗体的布局情况。

(3) 设置对象属性

设置窗体上控件对象的属性,可以在属性窗口中进行。

单击窗体上的 Button1 命令按钮,使其处于选定状态,此时属性窗口中会自动显示该命令按钮的所有属性,在属性列表中,选定属性名 "Text",在右列中将默认值 "Button1" 改为 "运行",如图 1.11 所示。



图 1.10 在窗体上添加控件

按照上述方法,可以设置以下对象的属性。

● 设置窗体 Form1 的 Text (标题名) 属性为"一个简单程序"

● 设置按钮 Button2 的 Text(标题名)属性为"结束"

此时窗体的布局情况如图 1.9 所示。

(4) 编写程序代码, 建立事件过程

设计好用户界面后,还需要在程序中添加代码,才能实现相应的功能。

双击窗体上的按钮 Button1,可以切换到代码窗口,同时在代码编辑区中会打开 Button1_Click 事件过程模板,如图 1.12 所示。用户也可以通过其他方法(如选择"视图"菜单中的"代码" 命令)进入代码窗口,只是此时没有 Button1_Click 事件过程模板,需要从对象列表框中选择 "Button1",在过程列表框中选择单击事件(Click),这样才可以打开事件过程模板。

Butt	onl		- 🖗 Click	
Pub	lic Class Form1			
	Private Sub Button1_Click(ByVal	sender As System.Object,	ByVal e As System. EventA	rgs) Handles Button1.Click
	End Sub			
End	Class			

图 1.12 Button1_Click 事件过程模板

在 Button1_Click 事件过程模板中,各部分的含意如下。

① 关键字 Private Sub 和 End Sub 用于定义一个过程。

② Button1_Click 表示事件过程名。

③ 事件过程名后面有一对圆括号"()",其中包含两个参数 sender 和 e,最后是关键字 Handles 及其相关内容。

用户可在该事件过程的过程体中插入如下语句:

TextBox1.Text = "欢迎你来到 VB.NET 世界!"

该语句的含意是将右边的文字显示在文本框 TextBox1 中。

按上述同样的步骤,可以打开 Button2_Click 事件过程模板,然后在该事件过程的过程体中插入 End 语句,该语句用于结束程序的运行。

此时代码窗口显示如图 1.13 所示。

@ But	ton2	-	- 🖗 Click	
⊟Pub	olic Class Form1			
	Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As Sys TextBox1.Text = "欢迎你来到VB.NET世界!" End Sub	Len	em.EventArgs) Handles Button1.Clic	≎k
End	Private Sub Button2_Click(ByVal sender As Object, ByVal e As System.Eve End End Sub I Class	ntA	tArgs) Handles Button2.Click	

图 1.13 例 1.1 的程序代码窗口

说明:

① 事件过程的模板是 VB.NET 系统为方便用户编程而自动提供的。初学者可以暂时不必关 心这些参数的含意,也不能对它们进行随意修改。

② 用户每输入完一行代码并按回车键时, VB.NET 能自动检查该行的语法错误。如果语句语法正确,则自动以不同的颜色显示代码的不同部分,并在运算符前后加上空格。

(5)保存项目

单击工具栏上的"全部保存"按钮,或选择"文件"菜单中的"全部保存"命令,打开"保

存项目"对话框,如图 1.14 所示。选择好保存"位置"(假设为"D:\VB\第1章"文件夹),"名称"栏中已显示先前设置的项目名称(如 WindowsApplication1),用户可以根据需要更改项目"名称"(本例改为"示例"),单击"保存"按钮,即可保存当前项目中所包含的全部文件(如窗体文件等)。

保存项目				? 🗙
名称(<u>N</u>):	示例			
位置(止):	D:\VB\第1章		•	浏览(<u>B</u>)
解决方案名称(M);	示例	创建解决方案的目录(D)		
		一添加到源代码管理(U)		
			保存(<u>S</u>)	取消

图 1.14 "保存项目"对话框

说明:本书中所有例题均为单一项目,所以不建议创建解决方案目录,也就是在"保存项目" 对话框中不选择"创建解决方案的目录"复选框。

	X
欢迎你来到VB	.NET世界!
运行	结束

(6)运行和调试程序

单击工具栏上的"启动调试"按钮,或选择"调试"菜 单中的"启动调试"命令,或按F5键,即可运行当前程序。 程序运行后,当用户单击"运行"命令按钮(Button1) 时,系统会自动执行 Button1_Click 事件过程,在窗体的文本 框上显示"欢迎你来到 VB.NET 世界!"字样,如图 1.15 所

图 1.15 单击"运行"按钮显示的情况示。单击"结束"命令按钮(Button2),则结束程序运行。

说明:程序运行过程中,单击窗体右上角的"关闭"按钮,或单击工具栏上的"停止调试" 按钮,也可以结束程序的运行。

如果程序运行出错,或未能实现所要求的功能,则需要进行修改,然后再次运行。对于大多数程序,通常要多次重复上述过程,此过程也称为"调试"。但要注意,在保存项目文件后对项目进行任何修改都需要再次保存项目文件。

1.4 项目的文件组成及常用操作

1.4.1 项目的文件组成简介

VB.NET 采用多种格式的文件来存储项目相关的 信息。例如,在例 1.1 中,当保存项目时,系统会在 用户指定的位置(如 D:\VB\第 1 章)下新建一个用项 目名"示例"命名的子文件夹(也称为解决方案文件 夹或程序文件夹),并在该文件夹下生成和保存多个文 件和子文件夹,主要涉及的文件如图 1.16 所示。当打 开一个项目时,该项目有关的所有文件同时装载。

初学者对这些文件夹和文件不必深究,但是对如 下文件应该有所了解。



图 1.16 一个简单程序的主要文件组成

① .sln 文件:该文件(如"示例.sln")称为 VB.NET 解决方案文件。VB.NET 中的开发工作 以解决方案的形式进行组织,每个解决方案包含一个或多个项目,存储定义一组项目关联、配置 等信息。

② .suo 文件: 该文件称为解决方案定义文件,存储一组项目中集成开发环境选项自定义的 信息。

③ .vbproj 文件: 该文件称为 VB.NET 项目文件,存储一个项目的相关信息,如窗体、类引用等。

④ .vb 文件: 该文件称为代码模块文件,也称为 VB.NET 源文件。在 VB.NET 中,所有包含 程序代码的源文件均以.vb 作为扩展名。例如,Form1.vb 存放用户在窗体 Form1 中所写的程 序代码。

⑤ .resx 文件: 该文件称为资源文件, 主要存放启动指定窗体的素材, 如图片、图标等。

⑥ .exe 文件(可执行文件): 在程序正常运行调试过之后, VB.NET 会自动生成一个与项目 同名的可执行文件(如"示例.exe"),该文件保存在项目所在文件夹下的"bin\Debug"子文 件夹中。

1.4.2 项目的常用操作

1. 关闭项目

执行"文件"菜单中的"关闭项目"命令,可将当前项目的相关文件关闭,如果项目还未保存,系统会提示用户保存或放弃该项目。

说明:如果执行"文件"菜单中的"关闭"命令,则只关闭当前打开的文件。

2. 打开项目

通常采用以下几种方法。

方法 1: 从起始页窗口的"最近使用的项目"中选择需要打开的项目。

方法 2: 执行"文件"菜单中的"打开项目"命令,打开"打开项目"对话框,然后选择扩展名为.sln 的文件(如"示例.sln"),再单击"打开"按钮。

方法 3:如果已经退出 VB.NET,在 Windows 环境下双击扩展名为.sln 的文件,可以打开相应的项目。

3. 对项目中某一文件重命名

要对项目中某一文件进行重命名,可以在解决方案资源管理器窗口中右击相应的文件,在快 捷菜单中选择"重命名"命令,直接输入新的名称,然后保存项目文件。

解决方案文件夹名的重命名,以及解决方案涉及的.sln 和.suo 文件的重命名,可以直接在 "Windows 资源管理器"中进行。

要注意的是,不要在 Windows 资源管理器下直接修改窗体文件等的文件名,更不要修改其 扩展名。

4. 项目的复制或删除

项目相关的信息(一系列文件夹和文件)都存放在项目所在的解决方案文件夹中,因此要删除或复制项目,只需在 Windows 资源管理器下按一般文件夹的删除或复制方法,对解决方案文件夹进行相应操作即可。

5. 项目的重命名

要修改项目的名称,如将"示例"项目名改为"例 1.1",可以通过对多个相关文件及文件夹 改名的方法来实现,但操作比较烦琐。

一种简单可行的方法是,创建一个新项目,选择"项目"菜单中的"添加现有项"命令,在 "添加现有项"对话框中查找用来复制的项目所在的解决方案文件夹(如"示例"文件夹),打开 该文件夹,选择如图 1.17 所示的相关文件(.resx 和.vb 文件)后,单击"添加"按钮,以替换新 建项目的原有同名文件。最后以新的项目名(如"例 1.1")保存即可。

组织 新建文件	挟			· #	
>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>	^	名称	修改日期	类型	大小
→ 音乐		🗼 bin	2017/8/18 21:38	文件夹	
No. 1 Steen		📕 My Project	2017/8/18 21:38	文件夹	
「「「「「」」」」」」」」」」」」」」」」」」」」」」」」」」」」」」」」		📕 obj	2017/8/18 21:38	文件夹	
● 本地磁館 (C:)		Form1.Designer.vb	2017/8/18 9:11	Visual Basic Source file	4 K
→ 本地磁盤 (D:)		Ja Form1.resx	2017/8/18 9:11	.NET Managed Reso	6 K
→ 中地磁量 (E:)		Di Form1.vb	2017/8/18 9:11	Visual Basic Source file	1 K
😧 网络	н				

图 1.17 选择用来添加(替换)的文件

习题 1

一、 单诜题

1.	要设置窗体上各控件的属性,一般可	在中进行。	
	A) 代码窗口	B)解决方案资源管理	提器窗口
	C) 属性窗口	D)窗体窗口	
2.	在 VB.NET 中,编写程序代码应在_	中进行。	
	A)即时窗口 B)属性窗口	C)代码窗口	D)输出窗口
3.	编程人员可以从中选择所需的	控件放置在窗体上,再	按照设计要求对其属性进行
修改。			
	A)工具箱	B) 解决方案资源管理	提器窗口
	C) 菜单栏	D)工具栏	
4.	假设窗体上已有一个控件是活动的,	为了在属性窗口中设置	窗体的属性,预先要执行的
操作是	o		
	A)单击窗体上没有控件的地方	B)单击任一个控件	
	C)双击任一个控件	D)双击窗体上没有搭	这件的地方
5.	在代码窗口中,当从对象列表框中选	定了某一对象后,在	中会列出该对象的事件。
	A)属性窗口	B)过程列表框	
	C)工具箱	D)工具栏	
6.	在设计阶段中,从窗体窗口切换到代	码窗口,不可以采用	的方法。
	A)单击窗体		

B) 双击窗体 C) 单击解决方案资源管理器窗口中的"查看代码"按钮 D)单击代码窗口中任何可见部位 7. 在设计阶段中,从代码窗口切换到窗体窗口,可以采用。 A) 双击代码窗口 B) 单击代码窗口 C) 单击解决方案资源管理器窗口中的"显示所有文件" 按钮 D) 单击解决方案资源管理器窗口中的"视图设计器" 按钮 8. 在下面窗口中, ______可以查看与项目有关的所有文件。 A)工具箱窗口 B)属性窗口 C)解决方案资源管理器窗口 D)窗体窗口 9. 项目文件的扩展名是。 A) vb B) sln C) .exe D) .vbproj 10. 下列叙述中错误的是。 A) 使用"文件"菜单中的"打开项目"命令,可以打开一个已经创建的项目 B) 执行"文件"菜单中的"全部保存"命令,可以保存项目 C)保存项目时,VB.NET 将根据所提供的项目名称在指定的位置下建立一个用项目名 命名的子文件夹 D) 一个 VB.NET 项目存盘后形成一个文件 11. 下列窗口中, 不是可浮动、可停靠的窗口。 A)工具箱窗口 B)代码窗口 C) 解决方案资源管理器窗口 D) 属性窗口 12. 当需要上下文帮助时,先选择要帮助的主题(如 Private),然后按 键,就可以直接

打开与之相关的帮助内容。

A) Esc B) F1 C) F10 D) Enter

二、填空题

1. VB.NET 集成在_____中,用户可以充分利用所有.NET 平台特性。

2. VB.NET 的三种工作模式是 (2) 、 (3) 和 (4)。

3. 用 VB.NET 设计应用程序,大致上包括____(5)__和___(6)__两部分工作。

4. 当进入 VB.NET 集成开发环境时,发现没有显示"工具箱"窗口,应选择_____菜单中的____(8)___命令,使"工具箱"窗口显示出来,并最好将其窗口的属性设置为____(9)___状态。

5. 在设计阶段中,要选定窗体上多个控件,可以按住____键的同时依次单击各个 控件。

6. 要对选定的多个控件调整格式,如对齐、统一大小、调整间距等,可以使用<u>(11)</u>菜 单中的有关命令。

7. 要对选定的多个 TextBox 控件设置相同的字体,可以通过属性窗口的___(12) 属性进行设置。

8. 在设计阶段中,双击工具箱中的控件按钮,即可在窗体的<u>(13)</u>位置上设置控件; 当双击窗体上某个控件时,所打开的是<u>(14)</u>窗口。

9. 在解决方案资源管理器窗口中,单击_____按钮打开窗体窗口,单击_____按

钮打开代码窗口。

10. 解决方案文件的扩展名是___(17)__。

上机练习1

1. 认识 VB.NET 集成开发环境。

启动 VB.NET, 新建一个 Windows 窗体应用程序项目, 进入集成开发环境中, 进行以下操作。

(1) 找出以下部分:工具箱、解决方案资源管理器窗口、属性窗口、窗体窗口。

(2)关闭工具箱,再打开工具箱(使用工具栏操作或菜单操作)。

(3) 自动隐藏工具箱,再关闭工具箱的自动隐藏功能。

(4)关闭属性窗口,再打开属性窗口(使用工具栏操作或菜单操作)。

(5) 双击窗体 Form1 的空白处,打开代码窗口,显示 Form1 Load 事件过程模板。

(6) 在解决方案资源管理器窗口中,使用"视图设计器"和"查看代码"按钮,在窗体窗口 和代码窗口之间进行切换。

(7) 在窗口中央区域选项卡组中,单击 "Form1.vb[设计]"和 "Form1.vb" 选项卡,在窗体 窗口和代码窗口之间进行切换。

(8)执行"窗口"菜单中的"重置窗口布局"命令,可以将集成开发环境中的窗口布局恢复 到软件安装时的初始状态。

(9)设置工具箱窗口为"浮动"状态,再移动该窗口的位置,然后将"浮动"状态改为"停 靠"状态。

(10) 执行"文件"菜单中的"关闭项目"命令,关闭项目。

2. 编写一个 VB.NET 应用程序,程序界面由 1 个窗体、2 个文本框和 2 个命令按钮组成。 程序运行后,单击"显示"按钮后,在 2 个文本框中分别显示"Welcome"和"欢迎",如图 1.18 所示;单击"传递"按钮后,将第 2 个文本框中的内容"欢迎"传送到第 1 个文本框中,如 图 1.19 所示。



□ 上机练习1-2
 □ ▼
 次迎
 显示
 传递

图 1.18 单击"显示"按钮后的显示情况



按以下步骤进行操作。

(1) 启动 VB.NET, 创建一个默认名"WindowsApplication1"的新项目,同时系统自动提供 一个空白的 Windows 新窗体,默认名为 Form1。

(2)单击工具箱中的控件"TextBox",在窗体上按住鼠标左键拖动,添加文本框 TextBox1。 使用同样操作,在窗体上添加另一文本框 TextBox2。

单击工具箱中的控件 "Button",在窗体上按住鼠标左键拖动,添加命令按钮 Button1。使用同样操作,在窗体上添加另一命令按钮 Button2。

(3) 在属性窗口中设置以下对象的属性:

● 设置窗体 Form1 的 Text 属性值为"上机练习 1-2"

● 设置按钮 Button1 的 Text 属性值为"显示"

● 设置按钮 Button2 的 Text 属性值为"传递"

(4) 编写程序代码, 建立事件过程

双击窗体上的按钮 Button1,切换到代码窗口,然后在 Button1_Click 事件过程模板的过程体中插入如下语句:

TextBox1.Text = "Welcome"

TextBox2.Text="欢迎"

再打开 Button2_Click 事件过程模板,然后在该事件过程的过程体中插入如下语句:

TextBox1.Text = TextBox2.Text

该语句的含意是将文本框 TextBox2 中的文本内容传送给文本框 TextBox1。

此时代码窗口显示如图 1.20 所示。

Pub	ic Class Form1
	Private Sub Button1_Click (ByVal sender As System. Object, ByVal e As System. EventArgs) Handles Button1. Click
	TextBox1.Text = "Welcome"
	TextBox2.Text = "欢迎"
	End Sub
	Private Sub Button2_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles Button2.Click
	TextBox1. Text = TextBox2. Text
	End Sub
End	Class

图 1.20 第 2 题的程序代码窗口

(5) 保存项目

单击工具栏上的"全部保存"按钮,或选择"文件"菜单中的"全部保存"命令,打开"保存项目"对话框。在对话框中选择好保存"位置"(假设为"D:\VB\第1章"文件夹),在"名称" 栏中输入"上机练习1-2",单击"保存"按钮,即可保存当前项目中所包含的全部文件。

说明:本书各章上机练习题所建立的项目均假设保存在"D:\VB\第 x 章"文件夹下,项目名 采用"上机练习 n-m"(n 为章号, m 为题号)。

(6)运行和调试程序

单击工具栏上的"启动调试"按钮,或选择"调试"菜单中的"启动调试"命令,即可运行 当前程序。

程序运行后,当用户单击"显示"按钮时,系统执行 Button1_Click 事件过程,显示结果如图 1.18 所示;单击"传递"按钮时,系统执行 Button2_Click 事件过程,显示结果如图 1.19 所示。

单击窗体右上角的"关闭"按钮,或单击工具栏上的"停止调试"按钮,可以结束程序 的运行。