



高等学校计算机科学与技术教材

VB.NET程序设计 实验指导与习题测试

COMPUTER Science and Technology

□ 余青松 江 红 编著

- 原理与技术的完美结合
- 教学与科研的最新成果
- 语言精练，实例丰富
- 可操作性强，实用性突出

清华大学出版社 ●

通大学出版社

高等学校计算机科学与技术教材

VB.NET 程序设计

实验指导与习题测试

余青松 江 红 编著

清华大学出版社

北京交通大学出版社

• 北京 •

内 容 简 介

本书是与《VB.NET 程序设计》配套使用的实验指导和习题测试教材。内容包括实验指导篇、习题测试篇和习题解答篇三部分。

实验指导篇针对课程教学特点,精心设计了14次实验内容,以方便具体的教学实践安排。各实验内容中设计的实验案例经典、实用、可操作性强。

习题测试篇针对配套教程的每个章节,精心设计了选择题、填空题、思考题、实践题,读者通过习题测试,进一步检验和巩固相应章节的知识点,做到举一反三,融会贯通。

习题解答篇提供了每章习题测试的参考解答,并对习题中的难点进行了详细剖析,可供读者参考。

本书可作为高等学校各专业的程序设计教程,同时也可作为广大程序设计开发者、爱好者的自学参考书。

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 13501256678 13801310933

图书在版编目(CIP)数据

VB.NET 程序设计实验指导与习题测试 / 余青松, 江红编著. —北京: 清华大学出版社;
北京交通大学出版社, 2011.8
(高等学校计算机科学与技术教材)
ISBN 978-7-5121-0685-7

I. ①V… II. ①余… ②江… III. ①BASIC 语言-程序设计-高等学校-教学参考资料 IV. ①TP312

中国版本图书馆 CIP 数据核字 (2011) 第 163448 号

责任编辑: 谭文芳

出版发行: 清华大学出版社 邮编: 100084 电话: 010-62776969 <http://www.tup.com.cn>

北京交通大学出版社 邮编: 100044 电话: 010-51686414 <http://press.bjtu.edu.cn>

印刷者: 北京交大印刷厂

经 销: 全国新华书店

开 本: 185×260 印张: 15 字数: 440 千字

版 次: 2011 年 9 月第 1 版 2011 年 9 月第 1 次印刷

书 号: ISBN 978-7-5121-0685-7/TP·659

印 数: 1~3 000 册 定价: 26.00 元

本书如有质量问题, 请向北京交通大学出版社质监组反映。对您的意见和批评, 我们表示欢迎和感谢。

投诉电话: 010-51686043, 51686008; 传真: 010-62225406; E-mail: press@bjtu.edu.cn。

编 者 的 话

本书是与《VB.NET 程序设计教程》配套使用的实验指导和习题测试教材，也可单独作为实践和测试教程或其他 VB.NET 程序设计教程的配套用书。

程序设计是大专院校计算机、电子信息、工商管理等相关专业的必修课程，VB/VB.NET 系列程序设计语言有着深厚的群众基础，是一门使用最广泛的程序设计语言。国内外众多的大专院校采用 VB/VB.NET 系列作为程序设计课程的必修语言。

本书主要基于 Visual Studio 2010/.NET Framework 4.0 开发和运行环境，讲述 VB.NET 10.0 的语言基础知识，以及使用 VB.NET 10.0 语言的实际开发应用实例。

本书内容包括实验指导篇、习题测试篇和习题解答篇。

实验指导篇针对课程教学特点，精心设计了 14 次实验内容，以方便具体的教学实践安排。具体包括：VB.NET 开发环境及程序设计基础、程序流程控制、数组、类的设计和实现、结构和枚举、多线程、泛型、特性、语言集成查询（LINQ）、文件和流 IO、集合和数据结构、数据库访问、Windows 窗体应用程序开发、ASP.NET Web 应用程序开发等。

习题测试篇针对配套教程的每个章节，精心设计了大量的选择题、填空题、思考题、实践题，读者通过习题测试，进一步检验和巩固相应章节的知识点，做到举一反三，融会贯通。

习题解答篇提供了每章习题测试的参考解答，并针对习题中的重点和难点，进行了详细剖析，可供读者参考。

本书涉及的源程序代码和相关素材，以及供教师参考的教学电子文稿均可以通过北京交通大学出版社网站（<http://press.bjtu.edu.cn>）的下载栏目下载，也可以通过图书支持网站 <http://www.gettop.net> 下载，还可以直接与作者联系 hjiang@cc.ecnu.edu.cn。

本书由华东师范大学余青松和江红共同编写。感谢蒲鹏老师、刘珏老师、刘艳老师和朱晴婷老师认真审阅了书稿，并提出了宝贵的修改意见和建议。

由于时间和编者学识有限，书中不足之处在所难免，敬请诸位同行、专家和读者指正。

编 者

2011 年 6 月

目 录

实验指导篇

实验 1 VB.NET 开发环境及程序设计基础	1
实验 2 VB.NET 语言基础	9
实验 3 程序流程控制 (1)	14
实验 4 程序流程控制 (2)	23
实验 5 数组	33
实验 6 类的设计和实现	45
实验 7 结构和枚举	54
实验 8 多线程、泛型和特性	62
实验 9 语言集成查询(LINQ)	70
实验 10 文件和流 IO	77
实验 11 集合和数据结构	92
实验 12 数据库访问	105
实验 13 Windows 窗体应用程序开发	115
实验 14 ASP.NET Web 应用程序开发	126

习题测试篇

第 1 章习题测试	139
第 2 章习题测试	141
第 3 章习题测试	143
第 4 章习题测试	154
第 5 章习题测试	164
第 6 章习题测试	169
第 7 章习题测试	177
第 8 章习题测试	184
第 9 章习题测试	186
第 10 章习题测试	192
第 11 章习题测试	194

第 12 章习题测试 199

第 13 章习题测试 202

第 14 章习题测试 206

第 15 章习题测试 208

第 16 章习题测试 210

习题测试参考解答 213

参考文献 230

实验指导篇

实验 1 VB.NET 开发环境及程序设计基础

实验目的

- ☞ 掌握使用命令行开发简单的 VB.NET 应用程序;
- ☞ 掌握使用 Visual Studio 编写控制台应用程序;
- ☞ 掌握 Visual Studio 环境下程序的跟踪调试;
- ☞ 了解 Visual Studio 在线帮助的使用;
- ☞ 掌握应用程序命令行参数的使用;
- ☞ 掌握使用 Visual Studio 编写 Windows 窗体应用程序;
- ☞ 掌握 Label、TextBox、Button 控件的应用。

实验内容

【实验 1-1】 使用命令行开发简单的 VB.NET 应用程序。

实验要求: 使用记事本编写“Hello World!” VB.NET 应用程序, 使用 VB.NET 命令行编译程序 vbc.exe 编译程序并运行测试。

操作步骤:

(1) 启动记事本, 输入如下代码, 并保存为 C:\VB.NETLAB\LAB01\Hello.vb。

```
' LAB01/Hello.vb ' A "Hello World!" program
' compile: vbc Hello.vb -> Hello.exe
Module Module1
    Sub Main()
        Console.WriteLine("Hello World!")
        Console.ReadKey()
    End Sub
End Module
```

(2) 启动 Visual Studio 2010 命令提示并进入 C:\VB.NETLAB\LAB01 文件夹。执行【开始】|【所有程序】|【Microsoft Visual Studio 2010】|【Visual Studio Tools】|【Visual Studio 命令提示(2010)】菜单命令, 进入 Visual Studio 2010 命令提示状态, 并输入命令: cd \VB.NETLAB\LAB01, 并按 Enter 键, 将当前目录切换到实验一的实验目录, 如图 1-1 所示。

(3) 编译 Hello.vb 程序。在“C:\VB.NETLAB\LAB01”命令提示状态, 输入命令: vbc Hello.vb, 并按 Enter 键, 如图 1-2 所示, 编译 Hello.vb 程序。

(4) 运行 Hello.exe 程序。在“C:\VB.NETLAB\LAB01”命令提示状态, 输入命令: Hello, 并按 Enter 键, 如图 1-2 所示, 运行 Hello.exe 程序, 观测运行效果。

【实验 1-2】 Visual Studio 的基本使用。

实验要求: 使用 Visual Studio 编写控制台应用程序 HelloConsole。根据命令行参数 (如 zhangsan), 输出: “您好! zhangsan”。



图 1-1 切换到实验一的实验目录

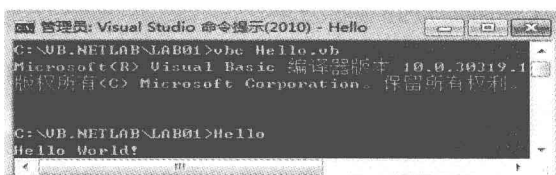


图 1-2 编译、运行 Hello.vb 程序

操作步骤:

(1) 打开 Visual Studio 2010。执行【开始】|【所有程序】|【Microsoft Visual Studio 2010】|【Microsoft Visual Studio 2010】菜单命令，打开 Visual Studio 2010。

(2) 创建控制台应用程序 HelloConsole。通过菜单【文件】|【新建项目】，打开【新建项目】对话框，选择“控制台应用程序”模板，并在“名称”文本框中输入项目名称：HelloConsole，如图 1-3 所示。单击【确定】命令按钮，创建控制台应用程序解决方案和项目。



图 1-3 创建控制台应用程序 HelloConsole

(3) 输入处理程序。在 Module1.vb 中添加如下粗体代码：

```
Module Module1
    Sub Main(ByVal args() As String)
        If (args.Length = 0) Then
            Console.WriteLine("请输入您的姓名作为参数!")
        Else
            Console.WriteLine("您好! " + args(0))
        End If
        Console.ReadKey()
    End Sub
End Module
```



图 1-4 提示用户输入姓名作为参数

(4) 编译运行。单击工具栏上的“启动调试”按钮，或者按“启动调试”的快捷键 F5 运行并测试该应用程序。运行效果如图 1-4 所示，提示用户输入姓名作为参数。

(5) 在 Visual Studio 环境输入命令行参数并调试运行。

鼠标右击解决方案资源管理器的项目 **HelloConsole**，执行相应快捷菜单中的【属性】命令，单击左列的【调试】设置，在“启动选项”的“命令行参数”中输入“zhangsan”，如图 1-5 所示。单击“关闭”按钮，关闭属性设置。按 F5 键再次运行并测试应用程序。运行效果如图 1-6 所示。



图 1-5 输入命令行参数



图 1-6 提供命令行参数的运行效果

(6) 保存项目。通过菜单命令【文件】|【保存全部】，或者单击 Visual Studio 工具栏中的“保存全部”按钮，均可打开【保存项目】对话框，利用【浏览】命令按钮，选择要保存的位置，单击【保存】命令按钮，如图 1-7 所示，系统保存项目到指定的位置。

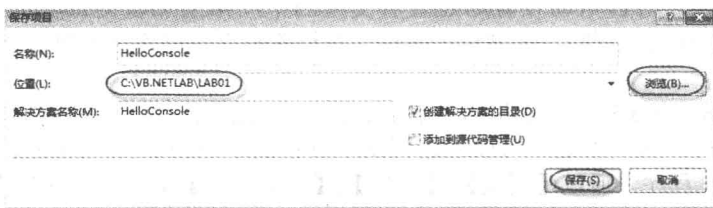


图 1-7 【保存项目】对话框

【实验 1-3】 Visual Studio 环境下程序的跟踪调试。

实验要求：熟悉 Visual Studio 环境下程序的跟踪调试功能，包括设置断点并进行跟踪调试等。

操作步骤：

(1) 打开 Visual Studio 2010，创建控制台应用程序 TraceDebug。

(2) 输入处理程序。在 **Module1.vb** 中添加如下粗体代码（注意：每条语句末故意输入“;”，制造 5 个编译错误）：

```
Module Module1
```

```
Sub Main()
```

```
    Dim a As Integer = 20;
```

```
    Dim b As Integer = 5;
```

```
    Dim c As Integer = 100 / a + b;
```

```
    Console.WriteLine(c);
```

```
    Console.ReadKey();
```

```
End Sub
```

```
End Module
```

(3) 编译程序。执行【生成】|【生成 TraceDebug】菜单命令编译程序，【错误列表】窗口中将列出如图 1-8 所示的 5 个错误信息，表明源代码中第 3~7 行存在错误。

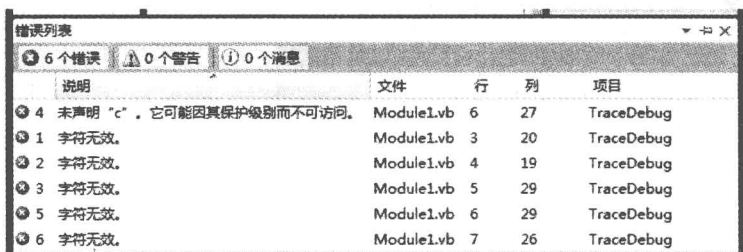


图 1-8 错误列表信息

(4) 修正编译错误。依次双击【错误列表】窗口中所列出的错误信息，光标将自动定位到出错的行列位置，依次删除“;”，改正 5 个编译错误。

(5) 编译运行。按 F5 键运行测试该应用程序。运行结果为：10。如果程序完全正确，Visual Studio 窗口底部的状态栏左侧将显示“生成成功”的提示信息。

(6) 运行时错误。故意将 Module1.vb 的 Main() 方法中的“Dim a As Integer = 20”改为“Dim a As Integer = 0”。按 F5 键运行测试该应用程序。编译通过，但是将出现如图 1-9 所示的运行时错误：“算术运算导致溢出(确保不会被零除)”。关闭当前运行窗口(也可以单击工具栏上的“停止调试”按钮)，否则运行调试状态将无法修改源代码)，再将“Dim a As Integer = 0”改回“Dim a As Integer = 20”。按 F5 键运行测试该应用程序，再次得到正确的运行结果“10”。

(7) 设置断点。假设本程序其实是计算“100/(a+b)”而不是程序设计时的“100/a+b”，这将是一个逻辑错误。可以借助设置断点查看变量的中间结果加以调试解决。在“Dim a As Integer = 20”所在行的左侧断点区域单击鼠标，在设置断点的位置会出现一个红点，如图 1-10 所示。

(8) 启动调试。按 F5 键启动调试，调试器将在断点处停止并获得当前窗口焦点。

(9) 单步执行。从断点处，可以通过调试工具栏的逐语句执行()、或者逐过程执行()，单步运行调试程序。通过【调试】|【窗口】|【局部变量】菜单命令打开【局部变量】调试窗口，查看局部变量 a、b、c 在运行过程中值的变化，如图 1-10 所示。

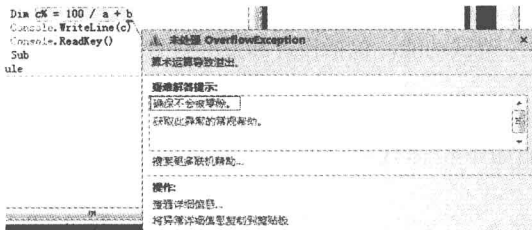


图 1-9 运行时错误

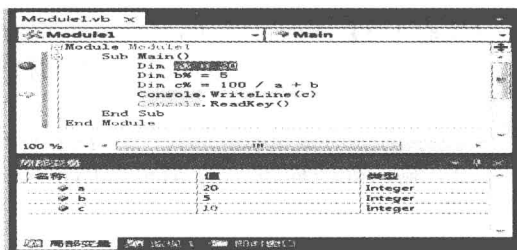


图 1-10 单步调试

(10) 清除断点并修正逻辑错误。单击调试工具栏中“停止调试”按钮)，停止调试，将“Dim c As Integer = 100 / a + b”改为“Dim c As Integer = 100 / (a + b)”。在“Dim a As Integer = 20”所在行的左侧断点区域再次单击鼠标，清除断点。按 F5 键运行测试该应用程序，得到正确的运行结果“4”。

(11) 保存项目。通过菜单命令【文件】|【保存全部】，或者单击 Visual Studio 工具栏中的“保存全部”按钮)，将项目 TraceDebug 保存到 C:\VB.NETLAB\LAB01 中。

【实验 1-4】 Visual Studio 在线帮助的使用。

实验要求：使用 Visual Studio 在线帮助功能。

操作步骤:

(1) 打开实验 1-2 的解决方案。执行【文件】|【打开项目】菜单命令,在随后的【打开项目】对话框中,选择 C:\VB.NETLAB\LAB01\HelloConsole\HelloConsole.sln 解决方案文件,打开实验 1-2 的解决方案。

(2) 使用上下文帮助。鼠标定位到要获得帮助的对象,例如关键字 WriteLine 中,按 F1 键,打开帮助窗口,显示 WriteLine 的帮助信息,如图 1-11 所示。



图 1-11 WriteLine 的帮助信息

(3) 使用搜索功能。在 Visual Studio 中按 F1 键,打开帮助窗口,在左上角的“搜索”文本框中,输入要搜索的关键字并按 Enter 键,可以显示与关键字相对应的主题,如图 1-12 所示。

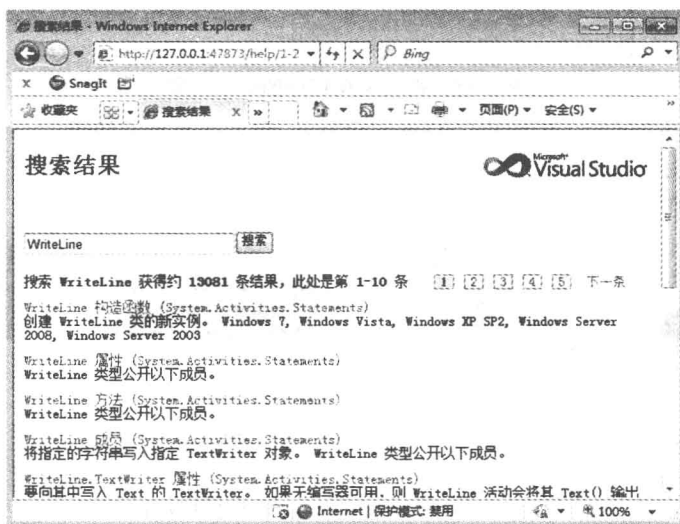


图 1-12 使用搜索功能

(4) 通过目录查看帮助信息。在 Visual Studio 中按 F1 键,打开帮助窗口,在左侧的“库主页”或者“相关链接”文本框中,选择需要查看的主题查看帮助文档内容。

【实验 1-5】创建控制台应用程序 ConsoleClass: 类、模块和对象的使用。

实验要求: 使用 Visual Studio 编写控制台应用程序 ConsoleClass,实现课本例 1.8 类和模块示例程序:在命名空间 VBBook.Chapter01 中定义类 Point;然后定义模块 TestModule,创建类 Point 的对象实例,最后输出平面上点的坐标值。运行效果如图 1-13 所示。

操作步骤: 程序代码参见图 1-14 所示。

注意: 本实验要正常运行,需要在解决方案资源管理器中,右击项目名称 ConsoleClass,执

行快捷菜单中的【属性】命令，将应用程序的“启动对象”改为“VBBook.Chapter01.Test Module”，如图 1-15 所示。

```
两个点的坐标分别为：
p1: x=0, y=0
p2: x=10, y=20
```

图 1-13 实验 1-5 运行效果

```
Module1.vb x
(代码)
Namespace VBBook.Chapter01
    Public Class Point
        Public x As Integer
        Public y As Integer
        Public Sub New(ByVal x As Integer, ByVal y As Integer)
            Me.x = x
            Me.y = y
        End Sub
    End Class
    Module TestModule
        Sub Main()
            Dim p1 As Point = New Point(0, 0) '点1
            Dim p2 As Point = New Point(10, 20) '点2
            Console.WriteLine("两个点的坐标分别为:")
            Console.WriteLine("p1: x={0}, y={1}", p1.x, p1.y)
            Console.WriteLine("p2: x={0}, y={1}", p2.x, p2.y)
            Console.ReadKey()
        End Sub
    End Module
End Namespace
100 %
```

图 1-14 实验 1-5 程序代码



图 1-15 改变应用程序的启动对象

【实验 1-6】创建控制台应用程序 ConsoleArgs：命令行参数的使用

实验要求：使用 Visual Studio 编写控制台应用程序 ConsoleArgs，实现课本例 1.10 命令行参数示例程序，输出命令行参数个数以及各参数内容。运行效果如图 1-16 所示。

```
参数个数 = 0
```

(a) 无参数

```
参数个数 = 2
Arg(0) = one
Arg(1) = two
```

(b) 2 个参数 (1)

```
参数个数 = 2
Arg(0) = one two
Arg(1) = three
```

(c) 2 个参数 (2)

```
参数个数 = 3
Arg(0) = a
Arg(1) = b
Arg(2) = c
```

(d) 3 个参数

图 1-16 实验 1-6 运行效果

操作步骤：请参考教材例 1.10 和实验 1-2 完成。程序代码如图 1-17 所示。注意：命令行参数之间使用空格分隔，如果是多个字符串作为一个参数，则使用双引号将其括起，如图 1-18 所示。

```
Module1
Main
Sub Main(ByVal cmdArgs() As String)
    Console.WriteLine("参数个数 = {0}", cmdArgs.Length)
    '使用for语句输出各参数值
    If cmdArgs.Length > 0 Then
        For argNum As Integer = 0 To cmdArgs.Length - 1
            Console.WriteLine("Arg({0}) = {1}", argNum, cmdArgs(argNum))
        Next argNum
    End If
    Console.ReadKey()
End Sub
End Module
100 %
```

图 1-17 实验 1-6 程序代码



图 1-18 在 Visual Studio 环境输入命令行参数

【实验 1-7】摄氏温度和华氏温度之间的转换。

实验要求：编写 Windows 窗体应用程序，使用 Label、TextBox、Button 控件，实现摄氏温度和华氏温度的转换（结果保留 2 位小数）。运行效果如图 1-19 所示。



图 1-19 实验 1-7 运行效果

摄氏温度 C 和华氏温度 F 之间的转换公式如下：

$$\begin{cases} F = \frac{9}{5}C + 32 \\ C = \frac{5}{9}(F - 32) \end{cases}$$

解决方案：本实验使用表 1-1 所示的 Windows 窗体控件完成指定的开发任务。

表 1-1 实验 1-7 所使用的控件属性及说明

控 件	属 性	值	说 明
Label1	Text	摄氏温度：	摄氏温度说明标签
Label2	Text	华氏温度：	华氏温度说明标签
TextBox1	Name	TextBoxC	摄氏温度文本框
TextBox2	Name	TextBoxF	华氏温度文本框
Button1	Name	CtoF	摄氏转为华氏命令按钮
	Text	=>	
Button2	Name	FtoC	华氏转为摄氏命令按钮
	Text	<=	

操作步骤：

(1) 创建 Windows 窗体应用程序。启动 Visual Studio，通过菜单命令【文件】|【新建项目】，打开【新建项目】对话框，选择【Windows 窗体应用程序】模板；在【名称】处输入项目的名称“Temperature”。单击【确定】命令按钮，创建 Windows 窗体应用程序解决方案和项目。

(2) 窗体设计。从“工具箱”中分别将 2 个 Label 标签控件、2 个 TextBox 文本框控件、2 个 Button 按钮控件拖动到窗体上。参照表 1-1 和运行效果图 1-19, 分别在属性窗口中设置各控件的属性, 并在 Windows 窗体设计器适当调整这 6 个控件的大小和位置。

(3) 创建处理控件事件的方法。

① 生成并处理 CtoF_Click 事件。双击窗体中的“ $\Rightarrow$ ”按钮控件, 系统将自动生成“CtoF_Click”事件处理程序, 在其中加入如下粗体语句, 以实现摄氏温度到华氏温度的转换:

```
Private Sub CtoF_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles CtoF.Click
    Dim c as Double = 9 * Double.Parse(TextBoxC.Text) / 5 + 32
    TextBoxF.Text = Format(c, "0.00")
End Sub
```

② 生成并处理 FtoC_Click 事件。双击窗体中的“ $\Leftarrow$ ”按钮控件, 系统将自动生成“FtoC_Click”事件处理程序, 在其中加入如下粗体语句, 以实现华氏温度到摄氏温度的转换:

```
Private Sub FtoC_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles FtoC.Click
    Dim f as Double = 5 * (Double.Parse(TextBoxF.Text) - 32) / 9
    TextBoxC.Text = Format(f, "0.00")
End Sub
```

(4) 运行并测试应用程序。单击工具栏上的“启动调试”按钮, 或者按快捷键 F5 运行并测试应用程序。

(5) 保存项目。通过菜单命令【文件】|【全部保存】, 或者单击 Visual Studio 工具栏中的“全部保存”按钮, 均可打开【保存项目】对话框, 利用【浏览】命令按钮, 选择要保存的位置: C:\VB.NETLAB\LAB01。单击【保存】命令按钮, 系统保存项目到指定的位置。

实验 2 VB.NET 语言基础

实验目的

- 掌握常量和变量的使用;
- 掌握运算符和表达式的使用;
- 掌握 Math 类和数学函数的使用;
- 掌握 Random 类和随机函数的使用;
- 掌握日期时间的使用;
- 掌握 String 类的使用。

实验内容

【实验 2-1】 常量、变量、运算符、表达式和赋值语句的使用。

实验要求: 利用 TextBox 输入半径, 求对应的圆的周长、面积; 对应的球体的体积, 最后在 Label 中输出结果。运行效果如图 2-1 所示。

解决方案: 本实验使用表 2-1 所示的 Windows 窗体控件完成指定的开发任务:

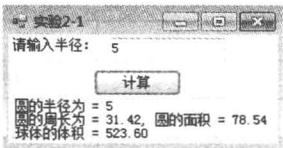


图 2-1 实验 2-1 运行效果

表 2-1 实验 2-1 所使用的控件属性及说明

控 件	属 性	值	说 明
Label1	Text	请输入半径:	提示输入标签
TextBox1			半径输入文本框
Button1	Text	计算	计算命令按钮
Label2	Text		结果显示标签

操作步骤:

- 创建 Windows 应用程序。启动 Visual Studio, 在 C:\VB.NETLAB\LAB02\sy2-1 文件夹中创建名为 Radius 的 Windows 窗体应用程序。
- 窗体设计。从“工具箱”中分别将 2 个 Label 控件、1 个 TextBox 控件及 1 个 Button 控件拖动到窗体上。参照表 2-1 和运行效果图 2-1, 分别在属性窗口中设置各控件的属性, 并在 Windows 窗体设计器适当调整这 4 个控件的大小和位置。
- 创建处理控件事件的方法。双击窗体上的“计算”按钮, 系统将自动生成“Button1_Click”事件处理程序, 在其中加入如下粗体语句, 以在结果显示 Label 中显示圆的周长、面积及球体的体积:

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    Const PI As Double = 3.14159
    Dim r, perimeter, area, volume As Double
    r = TextBox1.Text
    Label2.Text &= "圆的半径为 " & r & vbCrLf
    perimeter = 2 * PI * r
    area = PI * r * r
```

```

volume = 4 / 3 * PI * Math.Pow(r, 3)

Label2.Text &= "圆的周长为 = " & Format(perimeter, "0.00") & ", 圆的面积 = " _
    & Format(area, "0.00") & vbCrLf

Label2.Text &= "球体的体积 = " & Format(volume, "0.00") & vbCrLf

End Sub

```

(4) 运行并测试应用程序。单击工具栏上的“启动调试”按钮，或者按快捷键 F5 运行并测试应用程序。

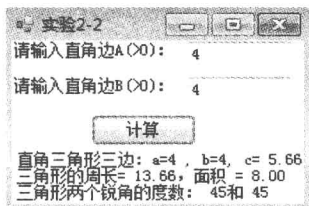


图 2-2 实验 2-2 运行效果

【实验 2-2】 Math 类和数学函数的使用。

实验要求：输入直角三角形的两个直角边，求三角形的周长和面积，以及两个锐角的度数（保留到整数部分）。运行效果如图 2-2 所示。

解决方案：

(1) 本实验使用表 2-2 所示的 Windows 窗体控件完成指定的开发任务：

表 2-2 实验 2-2 所使用的控件属性及说明

控 件	属 性	值	说 明
Label1	Text	请输入直角边 A(>0):	提示输入标签
TextBox1			直角边 A 输入文本框
Label2	Text	请输入直角边 B(>0):	提示输入标签
TextBox2			直角边 B 输入文本框
Button1	Text	计算	计算命令按钮
Label3	Text		结果显示标签

(2) Math.Asin 函数返回正弦值为指定数字的弧度；Math.Acos 函数返回余弦值为指定数字的弧度。

(3) 将弧度转换为角度的公式为：角度 = $\frac{\text{弧度} * 180}{\pi}$ 。

(4) 在 Form1.vb 源代码顶端添加“Imports System.Math”，将 System.Math 命名空间导入项目。

操作步骤：

(1) 创建 Windows 应用程序。启动 Visual Studio，在 C:\VB.NETLAB\LAB02\sy2-2 文件夹中创建名为 Maths 的 Windows 窗体应用程序。

(2) 窗体设计。从“工具箱”中分别将 3 个 Label 控件、2 个 TextBox 控件 1 个 Button 控件拖动到窗体上。参照表 2-2 和运行效果图 2-2，分别在属性窗口中设置各控件的属性，并在 Windows 窗体设计器适当调整这 6 个控件的大小和位置。

(3) 创建处理控件事件的方法。双击窗体上的“计算”按钮，系统将自动生成“Button1_Click”事件处理程序，在其中加入如下粗体语句，以在结果显示 Label 中显示直角三角形的三条边、周长和面积、两个锐角的度数：

```

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    Dim a, b, c, p, h, area As Double
    a = TextBox1.Text           '直角边 A
    b = TextBox2.Text           '直角边 B
    c = Sqrt(a ^ 2 + b ^ 2)     '斜边 C
    Label3.Text = "直角三角形三边: a=" & a & ", b=" & b & ", c=" & Format(c, "0.00") & vbCrLf
    p = a + b + c               '周长

```

```

h = p / 2
area = Sqrt(h * (h - a) * (h - b) * (h - c))
Label3.Text &= "三角形的周长=" & Format(p, "0.00") & _
    ", 面积 =" & Format(area, "0.00") & vbCrLf '面积
Dim sinA As Double = a / c
Dim aAngle As Double = Round(Asin(sinA) * 180 / PI, 0) '锐角 A
Dim cosB As Double = a / c
Dim bAngle As Double = Round(Acos(cosB) * 180 / PI, 0) '锐角 B
Label3.Text &= "三角形两个锐角的度数: " & aAngle & "和 " & bAngle & vbCrLf
End Sub

```

(4) 运行并测试应用程序。单击工具栏上的“启动调试”按钮, 或者按快捷键 F5 运行并测试应用程序。

【实验 2-3】 Random 类和随机函数的使用。

实验要求：随机生成一个 3 位数（整数），将其各位数字反转后输出。运行效果如图 2-3 所示。

解决方案：本实验使用表 2-3 所示的 Windows 窗体控件完成指定的开发任务。

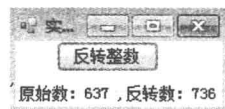


图 2-3 实验 2-3 运行效果

表 2-3 实验 2-3 所使用的控件属性及说明

控 件	属 性	值	说 明
Button1	Text	反转整数	反转整数命令按钮
Label1	Text		结果显示标签

操作步骤：

(1) 创建 Windows 应用程序。启动 Visual Studio，在 C:\VB.NETLAB\LAB02\sy2-3 文件夹中创建名为 Reverse 的 Windows 窗体应用程序。

(2) 窗体设计。从“工具箱”中分别将 1 个 Button 控件和 1 个 Label 控件拖动到窗体上。参照表 2-3 和运行效果图 2-3，分别在属性窗口中设置各控件的属性，并在 Windows 窗体设计器适当调整这 2 个控件的大小和位置。

(3) 创建处理控件事件的方法。双击窗体上的“反转整数”按钮，系统将自动生成“Button1_Click”事件处理程序，在其中加入如下粗体语句，以在结果显示 Label 中显示随机生成的 3 位数及其各数字反转后的结果：

```

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    Dim a, b As Integer
    Dim rNum As Random = New Random()
    a = rNum.Next(100, 1000)
    b = (a Mod 10) * 100 + (a \ 10 Mod 10) * 10 + a \ 100
    Label1.Text = "原始数: " & a & ", 反转数: " & b
End Sub

```

(4) 运行并测试应用程序。单击工具栏上的“启动调试”按钮, 或者按快捷键 F5 运行并测试应用程序。

【实验 2-4】 DateTime 类常用方法和属性的使用。

实验要求：打印当年 1 月份~12 月份的日历。运行效果如图 2-4 所示。