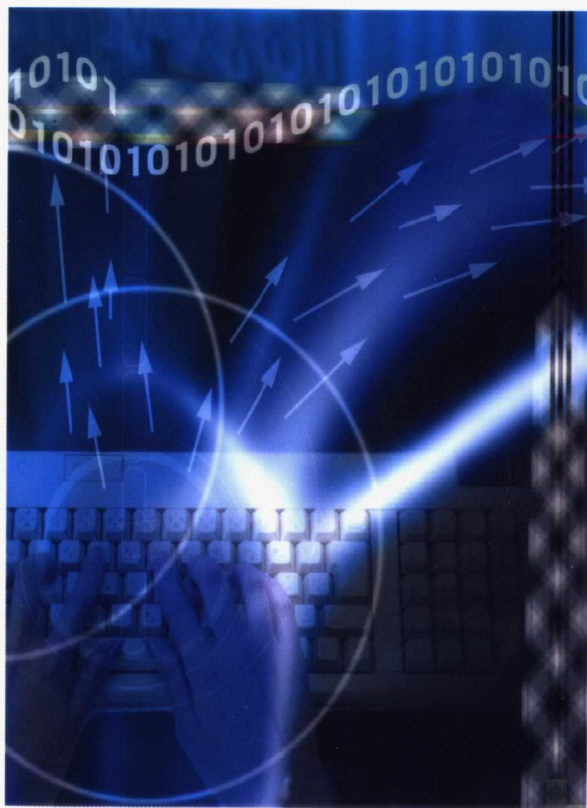


Visual Basic 程序设计教程

- ◆ Visual Basic .NET 2003 编程基础
- ◆ 面向对象编程
- ◆ Windows 程序设计
- ◆ Visual Basic .NET 2003 常用控件
- ◆ 组件实现技术
- ◆ 多线程编程技术
- ◆ 图形设计技术
- ◆ Web 程序技术
- ◆ 数据访问技术



彭琿 罗强 等编著



清华大学出版社

高等院校计算机应用技术系列教材

Visual Basic 程序设计教程

彭 琨 罗 强 等编著

清华大学出版社

北 京

内 容 简 介

Visual Basic .NET 是目前使用最广泛的编程语言之一,它并不是 Visual Basic 的简单升级,而是实现了编程方式的重大飞跃:它的一切都以类的形式实现。Visual Basic .NET 已经成为一种真正意义上的面向对象的编程语言了。全书共分 11 章,内容分别涉及 Visual Basic .NET 2003 的开发环境、语言及编程基础、Web 程序设计技术、数据访问技术等。

本书从传统编程方式入手,逐步向读者讲述 Visual Basic .NET 的应用方法。每章以理论为基础,配合具体的应用实例,内容丰富,结构清晰,语言简练,叙述深入浅出,具有很强的实用性。本书适合作为高等院校计算机程序设计的教材和参考书,同时也非常适合广大初、中级编程爱好者自学参考。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

Visual Basic 程序设计教程/彭琤,罗强 等编著. —北京:清华大学出版社,2004
(高等院校计算机应用技术系列教材)

ISBN 7-302-08240-5

I. V… II. ①彭… ②罗… 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 017351 号

出 版 者:清华大学出版社 地 址:北京清华大学学研大厦

<http://www.tup.com.cn> 邮 编:100084

社 总 机:010-62770175 客户服务:010-62776969

组稿编辑:孟毅新

文稿编辑:许书明

封面设计:陈 伟

版式设计:康 博

印 刷 者:北京牛山世兴印刷厂

装 订 者:北京市密云县京文制本装订厂

发 行 者:新华书店总店北京发行所

开 本:185×260 印张:21 字数:498 千字

版 次:2004 年 4 月第 1 版 2004 年 4 月第 1 次印刷

书 号:ISBN 7-302-08240-5/TP·5947

印 数:1~4000

定 价:30.00 元

本书如存在文字不清、漏印以及缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770175-3103 或(010)62795704

前 言

Visual Basic .NET 2003 是 Visual Studio .NET 2003 的一个重要组成部分, 为快速构建面向 Microsoft Windows 和 Web 的应用程序提供了更容易、更高效的语言和工具。对现有的 Visual Basic 开发人员和 Microsoft .NET 开发环境中的新开发人员来说, Visual Basic .NET 2003 无疑是最理想的开发工具, 因为它提供了增强的可视化设计器、提高的应用程序性能和功能强大的集成开发环境(IDE), 使用户能够轻松走上应用程序开发的快车道。

本书共分 11 章, 第 1~3 章介绍了 Visual Basic .NET 2003 的开发环境和语言及编程基础知识; 第 4 章介绍了面向对象技术的程序开发方法; 第 5 章介绍了 Windows 程序设计的一般方法。第 6~7 章介绍了 Visual Basic .NET 2003 的控件和组件使用及创建方法; 第 8 章介绍了在 Visual Basic .NET 2003 中多线程技术使用方法; 第 9 章介绍了 Visual Basic .NET 2003 的图形设计技术; 第 10 章介绍了 Web 程序设计技术; 第 11 章介绍了 Visual Basic .NET 2003 中的数据访问技术。

本书面向广泛的编程爱好者, 采用由浅入深、循序渐进的讲述方法, 在基本编程理论的基础之上配合适当的程序示例, 努力做到理论与实际相结合, 结构安排合理, 具有很强的实用性, 特别适合作为通用教材。此外, 本书包含了适量的习题和上机操作题, 习题类型有填空题、选择题和问答题, 帮助读者在学习完每章后进行总结、复习。另外, 当读者通读完本书的学习内容后, 还可以通过附录中的综合测试题检测自己的学习情况。

本书是集体智慧的结晶, 除封面署名的作者外, 参加本书编写的还有祁春、张立浩、徐燕华、徐燕萍、孔祥亮、徐帆、钟岚、王静琪、孔祥丰、邱丽、王维、张雪琴等人。由于作者水平有限, 加之创作时间仓促, 本书不足之处在所难免, 欢迎广大读者批评指正。

作 者

2003 年 10 月

目 录

第 1 章	初识 Visual Basic .NET 2003	1
1.1	Visual Basic .NET 2003 概述	1
1.1.1	Visual Basic .NET 的发展历程	1
1.1.2	开发环境特性	2
1.1.3	语言特性	4
1.1.4	全新的程序运行方式	5
1.2	Visual Basic .NET 2003 开发环境	8
1.2.1	标题栏	9
1.2.2	菜单栏	9
1.2.3	工具栏	11
1.2.4	工具箱	12
1.2.5	其他窗口	12
1.2.6	自定义用户集成开发环境	13
1.3	从一个简单应用程序开始	14
1.3.1	程序设计思想	15
1.3.2	创建应用程序	15
1.3.3	设置对象属性	16
1.3.4	编写程序代码	17
1.3.5	测试应用程序	18
1.4	习题与上机操作	18
1.4.1	填空题	18
1.4.2	选择题	19
1.4.3	简答题	19
1.4.4	上机操作	19
第 2 章	Visual Basic .NET 2003 语言基础	20
2.1	数据类型	20
2.1.1	基本数据类型	20
2.1.2	自定义数据类型	23
2.1.3	数组	24
2.1.4	枚举类型	26
2.1.5	数据类型的转换	28

2.2	常量和变量	29
2.2.1	常量	30
2.2.2	变量	32
2.3	运算符	35
2.3.1	赋值运算符	36
2.3.2	算术运算符	36
2.3.3	比较运算符	37
2.3.4	串联运算符	39
2.3.5	逻辑运算符	39
2.3.6	运算符的优先级	41
2.3.7	运算符的有效组合	41
2.4	习题与上机操作	43
2.4.1	填空题	43
2.4.2	选择题	43
2.4.3	简答题	43
2.4.4	上机操作	44
第3章	Visual Basic .NET 2003 编程基础	45
3.1	程序流程控制	45
3.1.1	分支控制	45
3.1.2	循环控制	50
3.1.3	控制语句嵌套	55
3.1.4	With 语句	56
3.2	Visual Basic .NET 的过程	56
3.2.1	Sub 过程	57
3.2.2	Property 过程	59
3.2.3	过程的参数传递	61
3.3	Visual Basic .NET 的函数	63
3.3.1	Function 函数	63
3.3.2	数学函数	66
3.3.3	字符串函数	67
3.3.4	日期函数及其他一些函数	68
3.4	习题与上机操作	68
3.4.1	填空题	68
3.4.2	选择题	68
3.4.3	简答题	69
3.4.4	上机操作	69

第 4 章 面向对象编程	70
4.1 面向对象概述	70
4.1.1 类	71
4.1.2 对象	71
4.1.3 实例	72
4.1.4 消息	72
4.1.5 属性	72
4.1.6 方法	73
4.1.7 继承性	73
4.1.8 多态性和函数重载	74
4.1.9 封装	74
4.2 Visual Basic .NET 2003 面向对象程序设计	74
4.2.1 Visual Basic .NET 2003 的类	74
4.2.2 Visual Basic .NET 2003 的对象	86
4.3 事件与委托	90
4.3.1 事件	90
4.3.2 事件处理程序	92
4.3.3 委托和 AddressOf 运算符	94
4.4 接口	95
4.4.1 接口定义	95
4.4.2 接口实现	96
4.4.3 基于接口的多态性	98
4.4.4 类继承与接口继承	99
4.5 类库	104
4.5.1 创建类库项目	104
4.5.2 使用类库	105
4.6 继承的层次结构分析	106
4.6.1 可扩展性	106
4.6.2 选择方法的访问级别	107
4.6.3 部署后的类设计更改	107
4.7 习题与上机操作	108
4.7.1 填空题	108
4.7.2 选择题	108
4.7.3 简答题	109
4.7.4 上机操作	109

第 5 章 Windows 程序设计	110
5.1 Windows 窗体设计	110
5.1.1 窗体概念	111
5.1.2 创建窗体	111
5.1.3 窗体属性、事件和方法	113
5.1.4 创建特殊效果窗体	115
5.1.5 窗体对象	117
5.1.6 Windows 窗体继承	117
5.2 MDI 多文档程序设计	121
5.2.1 创建 MDI 父窗体	121
5.2.2 创建 MDI 子窗体	121
5.2.3 活动的 MDI 子窗口	123
5.2.4 排列 MDI 子窗体	124
5.3 菜单设计	125
5.3.1 菜单 MainMenu	125
5.3.2 菜单增强功能	129
5.3.3 上下文菜单 ContextMenu	130
5.3.4 ContextMenu 综合应用	132
5.4 工具栏	133
5.4.1 Toolbar 控件介绍	134
5.4.2 ImageList 组件	134
5.4.3 向工具栏添加按钮	135
5.4.4 定义“工具栏”按钮的图标	136
5.4.5 触发工具栏按钮的菜单事件	137
5.5 状态栏	137
5.5.1 StatusBar 控件	138
5.5.2 面板事件的处理	138
5.5.3 单击面板事件处理	139
5.5.4 设置状态栏面板的大小	140
5.5.5 在运行时更新状态栏信息	140
5.6 控制台应用程序	141
5.6.1 控制台应用程序的特点	141
5.6.2 建立控制台应用程序项目	142
5.6.3 控制台信息的输入与输出	143
5.7 习题与上机操作	144
5.7.1 填空题	144
5.7.2 选择题	145

5.7.3 简答题	145
5.7.4 上机操作	145
第 6 章 Visual Basic .NET 2003 的常用控件	146
6.1 窗体控件概述	146
6.1.1 向 Windows 窗体添加控件	146
6.1.2 在 Windows 窗体中使用的控件	147
6.2 常规控件	149
6.2.1 TextBox 控件	149
6.2.2 LinkLabel 控件	152
6.2.3 CheckedListBox 控件	154
6.2.4 DomainUpDown 控件	156
6.2.5 NotifyIcon 控件	156
6.2.6 HScrollBar 和 VScrollBar 控件	158
6.2.7 Panel 控件	159
6.3 RichTextBox 控件	160
6.3.1 确定 RichTextBox 中格式属性更改的时间	160
6.3.2 在 RichTextBox 控件中显示滚动条	161
6.3.3 使用控件显示 Web 样式的链接	162
6.3.4 为 RichTextBox 控件设置字体属性	162
6.4 对话框控件	163
6.4.1 创建对话框	164
6.4.2 显示 Windows 窗体的对话框	164
6.4.3 对话框中的输入操作	164
6.4.4 关闭对话框并保留输入信息	165
6.4.5 从对话框的父窗体检索信息	166
6.4.6 消息框	167
6.5 附加组件介绍	168
6.5.1 ErrorProvider 组件	168
6.5.2 HelpProvider 组件	169
6.5.3 窗体 ToolTip 组件	171
6.6 控件开发技术	172
6.6.1 控件开发基础知识	173
6.6.2 开发继承型控件	173
6.6.3 开发组合型控件	175
6.7 习题与上机操作	177
6.7.1 填空题	177

6.7.2 选择题	178
6.7.3 简答题	178
6.7.4 上机操作	178
第7章 组件实现技术	179
7.1 组件结构	179
7.1.1 实现组件功能的方法	179
7.1.2 组件和程序集	181
7.1.3 组件的初始化和终止	182
7.2 方法、属性和事件在组件中的实现	184
7.2.1 组件中方法的实现	184
7.2.2 组件中的属性实现	185
7.2.3 组件和自定义控件的事件实现	189
7.3 组件创作	190
7.3.1 创建项目	191
7.3.2 添加构造函数和终结程序	191
7.3.3 将属性添加到类中	193
7.3.4 测试组件	193
7.3.5 从组件显示窗体	194
7.4 组件中的异常处理	195
7.4.1 引发组件异常的最佳做法	195
7.4.2 从组件引发异常	196
7.4.3 在组件中处理异常的建议	198
7.4.4 在组件中处理异常	199
7.5 代码库	199
7.5.1 使用代码库应注意的问题	199
7.5.2 创建代码库	200
7.6 习题与上机操作	201
7.6.1 填空题	201
7.6.2 选择题	201
7.6.3 简答题	201
7.6.4 上机操作	201
第8章 多线程编程技术	202
8.1 多线程概述	202
8.1.1 线程与进程	203
8.1.2 处理线程	203
8.1.3 何时使用多线程	204

8.1.4	多线程的优点与缺点	204
8.1.5	线程池	205
8.1.6	线程处理在程序设计中的应用	207
8.1.7	启用线程	207
8.2	多线程的处理	208
8.2.1	多线程应用程序	208
8.2.2	多线程过程的参数和返回值	209
8.2.3	同步线程	211
8.3	多线程处理的综合应用	212
8.3.1	创建应用程序框架	212
8.3.2	定义在单线程上运行的方法	213
8.3.3	处理线程中的事件	215
8.3.4	启动调用方法的线程	215
8.3.5	停止线程	216
8.4	创作多线程组件	216
8.4.1	创建项目	217
8.4.2	从组件接受用户的输入信息	219
8.4.3	添加多线程处理功能	222
8.4.4	多线程中的封送处理	223
8.4.5	锁定线程	225
8.5	习题与上机操作	226
8.5.1	填空题	226
8.5.2	选择题	226
8.5.3	简答题	226
8.5.4	上机操作	227
第9章	图形设计技术	228
9.1	Graphics 对象	228
9.1.1	Paint 事件处理	229
9.1.2	CreateGraphics 方法	229
9.1.3	从 Image 对象创建	229
9.2	画笔、画刷和颜色	230
9.2.1	画笔	230
9.2.2	画刷	231
9.2.3	纯色	231
9.2.4	图案	232
9.3.5	颜色	232

9.3	图形设计	234
9.3.1	绘制线条或空心形状	234
9.3.2	绘制实心形状	234
9.3.3	绘制文本	235
9.3.4	呈现图像	236
9.3.5	图形设计示例	236
9.4	习题与上机操作	241
9.4.1	填空题	241
9.4.2	选择题	241
9.4.3	简答题	242
9.4.4	上机操作	242
第 10 章	Web 程序设计技术	243
10.1	Web 应用程序介绍	243
10.1.1	Web 窗体页	243
10.1.2	Web 窗体的组件	244
10.1.3	Web 窗体页功能	244
10.2	ASP.NET 介绍	246
10.2.1	ASP.NET 页框架和 Web 窗体页	246
10.2.2	XML Web services	246
10.2.3	状态管理功能与应用程序事件	247
10.2.4	编译、配置和部署	247
10.2.5	安全性、调试支持与优化	247
10.2.6	性能计数器支持	248
10.3	Web 窗体代码模型	248
10.3.1	ASP.NET Page 类	248
10.3.2	从 Page 类派生 Web 窗体	249
10.3.3	单文件 Web 窗体页	250
10.4	Web 窗体页处理	251
10.4.1	往返行程	251
10.4.2	重新创建页	252
10.4.3	事件驱动模型	252
10.4.4	Web 窗体处理的各个阶段	253
10.5	创建基本的 Web 窗体页	254
10.5.1	创建项目和窗体	254
10.5.2	添加控件和文本	256
10.5.3	创建事件处理程序	257

10.5.4	Web 窗体应用开发总结	258
10.6	习题与上机操作	258
10.6.1	填空题	258
10.6.2	选择题	258
10.6.3	简答题	259
10.6.4	上机操作	259
第 11 章	数据访问技术	260
11.1	数据访问技术概述	260
11.1.1	OLE DB	260
11.1.2	ADO	261
11.1.3	ADO.NET	261
11.1.4	如何选择 ADO.NET 或 ADO	261
11.1.5	如何选择 OLE DB	263
11.2	通过 ADO.NET 访问数据	264
11.2.1	ADO.NET 的设计目标	264
11.2.2	ADO.NET 结构	265
11.2.3	.NET 数据提供程序	267
11.2.4	ADO.NET DataSet	269
11.2.5	为 .NET 数据提供程序编写通用代码	270
11.3	窗体中的数据访问	271
11.3.1	创建项目和窗体	271
11.3.2	创建和配置数据集	272
11.3.3	添加显示控件	275
11.3.4	操作数据库	276
11.4	使用参数化查询	277
11.4.1	创建项目和窗体	277
11.4.2	创建和配置数据集	277
11.4.3	添加显示数据的控件	279
11.4.4	测试与功能完善	282
11.5	Web 数据访问的实现	282
11.5.1	创建项目和窗体	283
11.5.2	创建和配置数据集	283
11.5.3	添加 DataGrid 以显示数据	284
11.5.4	测试与总结	285
11.5.5	总结	285
11.6	ADO.NET 中的并发控制	285

11.6.1	ADO.NET 中的数据并发	285
11.6.2	ADO.NET 和 Visual Studio .NET 中的并发控制	286
11.6.3	处理并发异常示例	288
11.7	习题与上机操作	294
11.7.1	填空题	294
11.7.2	选择题	294
11.7.3	简答题	295
11.7.4	上机操作	295
附录 A	综合测试题(一)	296
附录 B	综合测试题(二)	298
附录 C	习题答案	301
附录 D	综合测试题部分参考答案	315

第1章 初识Visual Basic .NET 2003

Visual Basic .NET 2003 是 Visual Studio .NET 2003 的一个重要组成部分, 为快速构建面向 Microsoft Windows 和 Web 的应用程序提供了最容易、最高效的语言和工具。对现有的 Visual Basic 开发人员和 Microsoft .NET 开发环境中的新开发人员来说, Visual Basic .NET 2003 无疑是最理想的开发工具, 因为它提供了增强的可视化设计器、提高的应用程序性能和功能强大的集成开发环境(IDE), 使用户能够轻松走上应用程序开发的快车道。

教学目标

通过对本章的学习, 读者应了解 Visual Basic .NET 的发展历程、Visual Basic .NET 2003 的新特性, 以及使用 Visual Basic .NET 2003 进行程序设计的方法。

教学重点与难点

- Visual Basic 的发展历程
- Visual Basic .NET 2003 的新特性
- Visual Basic .NET 2003 的开发环境
- Visual Basic .NET 2003 的程序设计方法

1.1 Visual Basic .NET 2003 概述

Visual Basic .NET 已经成为全世界使用最为广泛的编程语言之一, 它可以使用户轻松地开发出具有 Windows 界面效果的应用程序。作为 Visual Basic .NET 的升级版, Visual Basic .NET 2003 在功能和性能上都得到了进一步提高和改善。

1.1.1 Visual Basic .NET 的发展历程

作为一种通用编程语言, Visual Basic .NET 经历了几代技术的发展过程。从 1964 年 Basic 语言诞生以来, Basic 已经历了 4 个发展阶段。第 1 代 Basic 指最早期(70 年代中期)的、只有十几个语句的 Basic, 常称为基本 Basic; 第 2 代 Basic 是指微机出现初期的 Basic(70 年代中期到 80 年代中期), 其代表为 GW-Basic 和 Microsoft 的 Basic(即 MS-Basic); 第 3 代 Basic 是在 80 年代中期出现的结构化的 Basic 语言, 其代表为 True Basic、Quick Basic、Turbo

Basic; 第4代 Basic 是在 Windows 环境下编程使用的 Basic(90 年代), 其代表为 Visual Basic、GFA-Basic、CA-REALIZER 等。

随着互联网络技术的不断发展, Visual Basic .NET 作为 Microsoft.NET 计划的重要组件将会被越来越多的人所接受, 而 .NET 所要让人们接受的就是多种编程语言可以融为一体, 共同为某个具体项目服务。从 Visual Basic .NET 2003 入手, 以后的编程模式已经不再是人们常常提问的“你掌握哪门编程语言”, 而是“你会编程吗”。

.NET 的核心内容就是搭建第 3 代互联网平台, 同时建立这个平台上的运营规则。这个网络平台将解决网站之间的协同合作, 从而最大限度地获取信息。在 .NET 平台上, 不同网站之间通过相关的协定联系在一起, 网站之间可以自动交流、协同工作, 以提供最全面地服务。Microsoft 的战略构想就是, 统一所有的网络开发语言, 在同一个平台上协同运行, 使其突破不同软硬件造成的程序适应性差的缺点, 最终实现互联网络的完美统一。

Visual Basic .NET 更加突出了软件的易用性和智能化, 学习和使用都更容易。新一代的 Visual Basic .NET 2003 是真正的面向对象的语言, 不再是基于对象的语言, 这个进步是非常重大的。同时 Microsoft 基于它的 Internet 战略为 Visual Basic .NET 2003 设计了更强大的 XML、HTML、Web 窗体和 Web 服务开发环境。总之, Visual Basic .NET 2003 已经是一种新的语言, 而不是一种语言的简单升级。

1.1.2 开发环境特性

Visual Basic .NET 2003 不再是单一的开发环境, 用户启动 Microsoft Visual Studio .NET 2003 时, 将显示如图 1-1 所示的提示界面, 并进入 Visual Studio .NET 2003 的开发环境。原先 Visual Basic、C++、C#、VJ 等各个独立的开发环境, 如今变成了 Visual Studio .NET 2003 中的各个不同的项目, 用户可以通过建立不同的项目来使用自己习惯的编程语言开发应用程序。

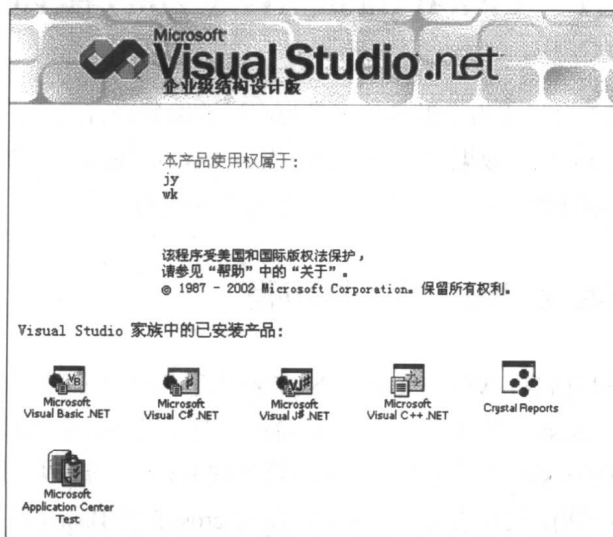


图 1-1 Visual Basic .NET 2003 提示界面

进入 Visual Studio .NET 2003 开发环境后,展现在用户面前将是 Visual Studio .NET 的“起始页”界面。它显示了最近打开过的项目,并提供了打开现有项目、创建新项目的选项(如图 1-2 所示),所有这些选项都以 HTML 链接形式显示。起始页是用 DHTML 形式编写的,用户可以定制起始页用来显示需要的选项、信息以及 HTML 链接。

起始页是 Microsoft 为 Visual Studio .NET 的集成开发环境定制的一个功能强大、内容丰富、连接快捷的联机帮助系统,即使是从没有使用接触过 Visual Studio .NET 的人也能通过该起始页面很快进入到 Visual Studio .NET 的世界中去。

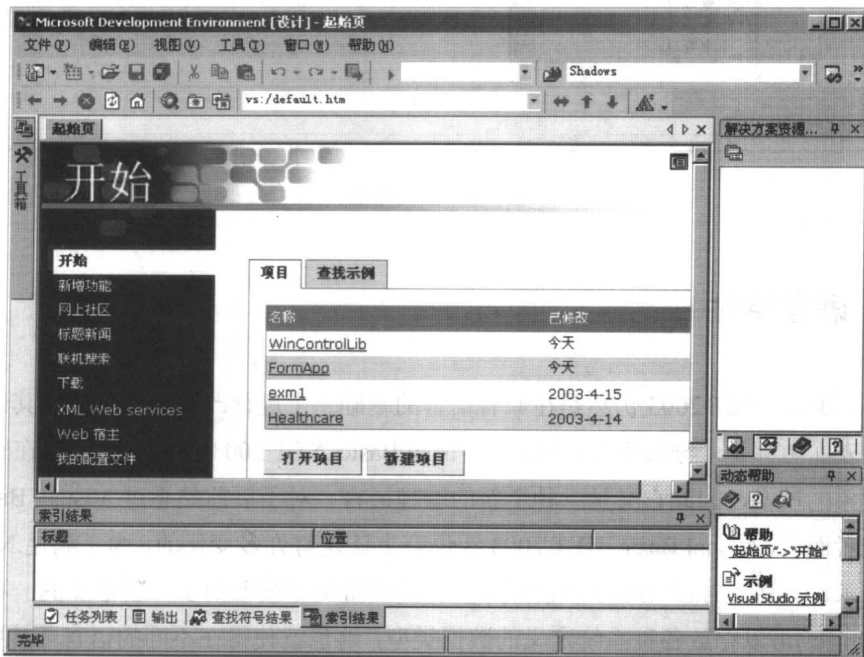


图 1-2 起始页

在起始页中,用户还可以看到其他一些窗口,例如解决方案资源管理器、动态帮助、索引结果等,通过这些窗口可以非常方便地转到相应的窗口或环境。当然,用户也可以关闭或者自行定义这些窗口,以此来满足自己的实际要求。因此,Visual Basic .NET 2003 给用户的最大感受就是以用户为中心,使非常枯燥、繁琐的编程工作变得十分简单,开发应用程序将是一件十分简单而有趣的事情。

在编程环境中(如图 1-3 所示),Visual Basic .NET 2003 提供了目前通用编程环境所支持的智能输入功能,也就是在用户输入一个关键字并加入符号“.”之后,开发环境会自动弹出一个下拉列表框,给出与该关键字相关联的子关键字列表供用户选择使用,而且当用户将光标定位到某个关键字时,开发环境还会在该关键字旁边给出它的语法规则和功能提示信息。即使用户对某个关键字掌握不十分熟悉,也能够通过这些提示信息了解它的具体使用方法,使得编写程序变得十分简便。