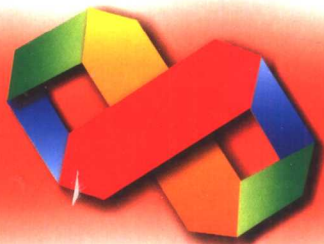




新版职业技能培训教材



中文版 Visual Basic.NET 实用教程

华创工作室 策划 郝佳波 黄仁祥 编著



西安电子科技大学出版社
<http://www.xduph.com>

TP312
1930

新版职业技能培训教材

中文版 *Visual Basic.NET*

实用教程

华创工作室 策划
郝佳波 黄仁祥 编著

西安电子科技大学出版社

2006

内 容 简 介

本书讲述面向对象的程序设计工具——Visual Basic.NET。通过图文并茂的形式和丰富详尽的实例，循序渐进、深入浅出地介绍了 Visual Basic.NET 的基础知识、面向对象编程的基本概念以及面向对象可视化程序设计的方法和开发技术。全书共 13 章，包括 Windows 编程、Web 编程、ASP.NET 以及数据库访问技术和实例等内容。

本书适合作为各类学校面向对象程序设计的教材，也适合作为 Visual Basic.NET 爱好者的自学用书。

图书在版编目(CIP)数据

中文版 Visual Basic.NET 实用教程 / 郝佳波等编著. —西安: 西安电子科技大学出版社, 2006.1
新版职业技能培训教材

ISBN 7-5606-1613-5

I. 中… II. 郝… III. BASIC 语言—程序设计—技术培训—教材
IV. TP312

中国版本图书馆 CIP 数据核字(2005)第 141841 号

策 划 李惠萍

责任编辑 邵汉平 李惠萍

出版发行 西安电子科技大学出版社(西安市太白南路 2 号)

电 话 (029)88242885 88201467 邮 编 710071

<http://www.xduph.com> E-mail: xdupfxb@pub.xaonline.com

经 销 新华书店

印刷单位 陕西天意印务有限责任公司

版 次 2006 年 1 月第 1 版 2006 年 1 月第 1 次印刷

开 本 787 毫米×1092 毫米 1/16 印 张 18.75

字 数 438 千字

印 数 1~6000 册

定 价 26.00 元

ISBN 7-5606-1613-5/TP·0925

XDUP 1905001-1

如有印装问题可调换

本社图书封面为激光防伪覆膜，谨防盗版。

前 言

在当今的信息化时代，电脑已经广泛应用于各个领域，而掌握可视化程序设计的方法则是显示应用能力的重要标志之一。**Visual Basic.NET** 是一种面向对象的程序设计工具，它采用了面向对象、事件驱动的程序设计模式，同时又保留了结构化程序设计语言的优点，其解决问题的方式更符合人们的思维习惯。所以，在各个领域其应用都是非常广泛的，许多计算机专业人员常利用它来开发应用软件。

Visual Basic.NET 是由 **Visual Basic 6.0** 升级得到的，这使得 **Visual Basic** 语言发生了革命性的变革，主要体现在 **Visual Basic** 真正成为了一种面向对象的编程语言。因此，本书以 **Visual Basic.NET** 应用为主线，重点讲解了计算机语言、程序设计的基本理论和常用算法，详细介绍了面向对象程序设计的概念、方法和控件，以及 **Web**、**ASP.NET** 和数据库的基本知识。全书共 13 章，包括 **Windows** 编程、**Web** 编程、**ASP.NET** 编程、数据库访问技术及实例等内容。本教材结构合理，内容精练，可操作性强。通过可视化界面设计和代码知识的有机结合，不仅能让读者深入了解各知识点，而且还能提高读者的实际操作能力和解决实际应用问题的能力。

本书虽经反复斟酌，但难免会有错误与疏漏之处，恳请广大读者批评指正。

编 者
2005 年 10 月

目 录

第 1 章 Visual Basic.NET 入门	1
1.1 Visual Basic 概述	1
1.1.1 Visual Basic 简介	1
1.1.2 Visual Basic 的发展过程	1
1.1.3 Visual Basic 版本简介	1
1.1.4 Visual Basic 的新特性	2
1.2 Visual Basic.NET 的安装和启动	4
1.2.1 安装 Visual Basic.NET 的软、硬件要求	4
1.2.2 Visual Basic.NET 集成开发环境(IDE)的安装	5
1.2.3 删除 Visual Studio.NET 的组件	6
1.2.4 Visual Basic.NET 的启动	6
1.2.5 第一个 Visual Basic.NET 程序	7
1.2.6 Visual Studio.NET 的退出	8
1.3 Visual Studio.NET 集成开发环境	8
1.4 Visual Basic.NET 帮助系统的使用	12
1.4.1 MSDN Library 在线帮助介绍	12
1.4.2 MSDN Library 在线帮助的使用	12
1.4.3 总结	13
习题	13
第 2 章 Visual Basic.NET 语言基础	14
2.1 源文件	14
2.2 标识符	14
2.3 关键字	15
2.4 命名空间	20
2.4.1 在程序中定义命名空间	20
2.4.2 在程序中使用命名空间的类	20
2.5 数据类型	22
2.5.1 基本数据类型	22
2.5.2 用户自定义类型	25
2.5.3 类型转换	25
2.6 常量	28
2.6.1 直接常量	28

2.6.2 符号常量	29
2.7 变量	29
2.8 元素的访问性	31
2.8.1 变量的作用域	32
2.8.2 静态变量	33
2.9 运算符和表达式	34
2.10 语句	41
2.11 语句的书写规则	42
2.12 常用函数	42
2.12.1 数学运算函数	43
2.12.2 字符串函数	44
2.12.3 日期和时间函数	46
2.12.4 随机数语句和函数	47
2.12.5 数值类型转换函数	47
习题	48
第3章 数组	50
3.1 数组的概念	50
3.1.1 数组的概述	50
3.1.2 数组的使用	51
3.1.3 有关数组类的属性及方法的使用	54
3.2 静态数组	56
3.2.1 数组的声明	56
3.2.2 数组的使用	57
3.2.3 静态数组的应用	58
3.3 动态数组	59
3.3.1 创建动态数组的方法	59
3.3.2 保留动态数组的内容	60
3.4 For Each...Next 语句	61
3.5 数组的基本操作	62
3.5.1 数组的引用	62
3.5.2 数组的复制	63
3.5.3 数组的输出	63
3.5.4 数组的初始化	64
3.6 数组的应用	64
3.6.1 统计字符	64
3.6.2 选择法排序	65
3.6.3 冒泡排序	67
习题	68

第 4 章 选择结构设计	69
4.1 选择结构概述.....	69
4.2 条件表达式.....	69
4.2.1 比较运算符与条件表达式.....	69
4.2.2 逻辑运算符与逻辑表达式.....	71
4.2.3 运算符的优先顺序.....	73
4.3 If 语句.....	73
4.3.1 简单分支结构.....	73
4.3.2 使用 If 函数.....	75
4.3.3 选择分支条件语句.....	75
4.3.4 If 语句的嵌套.....	76
4.4 多分支条件选择语句.....	77
4.5 End 语句.....	79
习题.....	79
第 5 章 循环结构设计	80
5.1 Do...Loop 循环语句.....	80
5.1.1 前测型 Do...Loop 循环.....	80
5.1.2 后测型 Do...Loop 循环.....	82
5.2 For...Next 语句.....	83
5.3 While...End While 循环.....	86
5.4 Exit 语句.....	87
5.5 多重循环.....	88
习题.....	89
第 6 章 过程	90
6.1 Sub 过程.....	90
6.1.1 Sub 过程的定义.....	90
6.1.2 参数声明.....	91
6.1.3 调用语法.....	91
6.2 Function 过程.....	92
6.2.1 Function 过程的定义.....	92
6.2.2 函数返回值.....	92
6.2.3 调用语法.....	93
6.3 Property 过程.....	94
6.3.1 属性声明.....	94
6.3.2 参数声明.....	95
6.3.3 调用语法.....	95
6.4 参数传递.....	96

6.4.1 参数传递的方式	96
6.4.2 数组参数	101
6.4.3 递归	104
习题	105
第 7 章 类	106
7.1 类与对象概述	106
7.1.1 对象的概念	106
7.1.2 类的概念	106
7.1.3 类与对象	107
7.2 Visual Basic.NET 中的类	107
7.3 类的创建	108
7.3.1 创建类	108
7.3.2 类关键字	109
7.3.3 类与名空间	109
7.3.4 创建方法	110
7.3.5 创建属性	111
7.3.6 默认的属性	111
7.3.7 重载方法	112
7.4 对象的生命周期	113
7.4.1 构造	113
7.4.2 对象的终止	114
7.5 继承	116
7.5.1 实现基本的继承	116
7.5.2 阻止继承	119
7.5.3 继承和辖域	119
7.5.4 Protected(保护)方法	120
7.5.5 重载方法	120
7.5.6 重载构造函数(Sub New)	123
7.6 共享及类成员	123
7.6.1 共享方法	123
7.6.2 共享变量	124
7.6.3 全局变量	125
7.7 事件和事件处理	126
7.7.1 事件	126
7.7.2 声明和引发一个事件的实例	127
7.7.3 事件处理的实例	129
7.8 接口	131
7.8.1 接口声明	132

7.8.2 重载方法	132
7.8.3 实现接口	132
7.8.4 实现多个接口	133
7.9 对象的处理	134
7.9.1 对象声明和实例化	134
7.9.2 取消引用对象	135
7.9.3 早期和晚期绑定	135
7.10 多态性	137
7.10.1 多态性的实现	138
7.10.2 用继承实现多态性	138
7.10.3 用接口实现多态性	139
习题	141
第 8 章 可视化编程的概念与步骤	142
8.1 可视化编程的基本概念	142
8.1.1 项目概述	142
8.1.2 类与对象	144
8.1.3 事件与程序	147
8.1.4 VB 的控件	147
8.1.5 代码窗口	148
8.2 Visual Basic 可视化编程的步骤	149
8.2.1 创建新的 Visual Basic.NET 项目	150
8.2.2 创建应用程序的用户界面	150
8.2.3 添加控件	151
8.2.4 设置属性	152
8.2.5 编写代码	153
8.2.6 保存项目	155
8.2.7 运行项目	155
8.2.8 创建可执行文件	156
习题	156
第 9 章 Visual Basic.NET 的常用控件	157
9.1 Label 控件	157
9.2 Button 控件	158
9.3 TextBox 控件	160
9.4 MainMenu 控件	163
9.5 RadioButton 控件	164
9.6 GroupBox 控件	165
9.7 ListBox 控件	166

9.8	CheckBox 控件	169
9.9	Timer 控件	170
9.10	PictureBox 控件	172
9.11	CheckedListBox 控件	173
9.12	ComboBox 控件	174
9.13	TreeView 控件	177
9.14	HScrollBar 控件和 VScrollBar 控件	179
9.15	ProgressBar 控件	180
9.16	ToolBar 控件	181
	习题	182
第 10 章	窗体和对话框编程	184
10.1	创建窗体	184
10.1.1	使用 Visual Basic.NET 创建窗体	184
10.1.2	添加事件处理代码	189
10.2	响应窗体事件	190
10.3	窗体之间的关系	191
10.3.1	设置窗体的 Parent 属性	192
10.3.2	设置窗体的 Owner 属性	193
10.4	控件和控件事件	194
10.5	窗体和控件布局	195
10.5.1	Anchor 属性	195
10.5.2	Dock 属性	196
10.6	通用对话框	198
10.6.1	颜色对话框	198
10.6.2	字体对话框	199
10.6.3	打开文件对话框	199
10.6.4	保存文件对话框	200
10.6.5	页面设置对话框	200
10.6.6	打印预览对话框	201
10.6.7	打印对话框	201
10.7	菜单	202
10.7.1	在窗体设计器中添加菜单	202
10.7.2	使用程序方式创建菜单	204
10.7.3	处理菜单事件	206
10.8	MDI 应用程序	206
10.8.1	父窗体和子窗体	206
10.8.2	创建窗体菜单	206
10.8.3	合并菜单	207

10.8.4 检测激活的 MDI 子窗体	208
10.9 打印	209
10.9.1 和打印任务相关的类	209
10.9.2 实现基本的打印功能	209
10.9.3 定义自己的打印文档	213
10.9.4 在打印前选择打印机	214
10.9.5 在打印之前进行页面设置	215
10.9.6 在打印之前进行打印预览	216
习题	217
 第 11 章 使用 ASP.NET 进行 Web 编程	218
11.1 ASP.NET 简介	218
11.1.1 什么是 ASP.NET	218
11.1.2 ASP.NET 的优点	218
11.2 ASP.NET Web 编程基础	219
11.2.1 Web 服务器和 IIS	219
11.2.2 HTML 语言	220
11.2.3 创建一个 Web 应用程序	223
11.3 ASP.NET Web 窗体介绍	226
11.3.1 什么是 ASP.NET Web 窗体	226
11.3.2 编写一个 Web 窗体页	226
11.3.3 使用 ASP <% %> 呈现块	227
11.3.4 ASP.NET 服务器控件介绍	228
11.3.5 处理服务器控件事件	230
11.3.6 代码隐藏(Code-Behind)的 Web 窗体	231
11.4 ASP.NET 控件	233
11.4.1 HTML 服务器控件	233
11.4.2 Web 服务器控件	233
11.5 Web 窗体语法	234
11.5.1 呈现代码语法	234
11.5.2 声明代码语法	234
11.5.3 ASP.NET 服务器控件语法	234
11.5.4 ASP.NET HTML 服务器控件语法	235
11.5.5 数据绑定语法	235
11.5.6 对象标记语法	236
11.5.7 服务器端注释语法	237
11.5.8 服务器端包含语法	237
11.6 ASP.NET 应用程序	237
11.6.1 什么是 ASP.NET 应用程序	237

11.6.2 ASP.NET 应用程序示例.....	239
习题	243
第 12 章 使用 ADO.NET 访问数据库.....	244
12.1 什么是 ADO.NET.....	244
12.2 ADO.NET 组件.....	244
12.3 .NET Framework 数据提供程序.....	247
12.4 ADO.NET DataSet.....	248
12.5 使用 ADO.NET 访问数据库.....	249
12.6 编写学生信息管理程序.....	258
习题	262
第 13 章 仿记事本程序.....	263
13.1 新建工程	263
13.2 设计记事本主窗体	263
13.3 设计打印类	280
13.4 设计“转到”对话框	283
13.5 设计“选项”对话框	285
13.6 编译并运行程序	288

第1章 Visual Basic.NET 入门

1.1 Visual Basic 概述

1.1.1 Visual Basic 简介

Basic 指的是 Beginners All-Purpose Symbolic Instruction Code 语言,它是在计算机科学技术领域中应用最为广泛的语言之一。Visual Basic 是美国 Microsoft 公司在 1991 年推出的编程语言。Visual Basic.NET(又称 Visual Basic 7.0 或 VB 7.0)是 Microsoft 公司在 Visual Basic 6.0 之后推出的最新版本,集成在 Visual Studio .NET 中,与 Visual C++ .NET 及 Visual C# 一起运行于“.NET”构架之上。

Visual 是可视化的意思,Visual Basic 的命名就表示它是一套可视化的程序开发工具。在设计过程中,可以用可视化方式去编排一个窗口程序在执行时的外观;同时,Visual Basic 也采用了易学易懂的 Basic 语言,并且把关于界面操作的细节部分事先封装起来,使得初学程序设计的读者可以轻而易举地写出心中想要的程序。

Visual Basic 是开发 Windows 应用程序、Web 应用程序、移动通信应用程序和 Office 应用程序最快捷和最有效率的开发工具之一。

1.1.2 Visual Basic 的发展过程

Visual Basic 是 Microsoft 公司为了简化 Windows 应用程序开发,于 1991 年在原 Basic 语言的基础上推出的新一代面向对象的程序设计语言,于 1992 年推出 VB 2.0,1993 年 4 月又推出 VB 3.0。随着 Windows 95 的发行,为适应 32 位操作系统的需要,又于 1995 年 8 月推出了开发 32 位应用程序的 VB 4.0。为了增强对 Internet 的支持和开发能力,于 1997 年推出了 VB 5.0。随着 Windows 98 的发行,1998 年又推出了功能更强、更完善的 VB 6.0。2000 年,Microsoft 推出 .NET 框架,Visual Basic 迎来了 .NET 时代。如今 .NET 框架还在发展,而 Visual Basic 也在继续发展,目前的最新版本是 Visual Basic.NET 2005。

1.1.3 Visual Basic 版本简介

自 Visual Basic 6.0 发行以来,包含 Visual Basic 的 Visual Studio 大致都有三种版本可以使用,每种版本都是为了满足特定的开发需求。Visual Studio .NET 也有三个版本:企业版、专业版和学院版。

1. Visual Studio.NET 企业版

Visual Studio.NET 企业版允许专业人员以小组的形式来创建强健的分布式应用程序。它包括专业版的所有特性,加上一系列的其他应用软件,如 Visual SourceSafe、Exchange Server、Commerce Server 等,使得开发人员可以为客户提供几乎所有的解决方案。

2. Visual Studio.NET 专业版

Visual Studio.NET 专业版向计算机专业人员提供了一套功能完整的工具,以方便用户开发应用程序。它除了包括 Visual Basic.NET 以外,还包括 Visual C++.NET、Visual C#、Visual J# 等开发语言。它可以创建 Windows 应用程序、Web 应用程序,创建并使用 XML Web Services,创建移动和智能手机应用程序等;还可以在 SQL Server 上创建表和视图。

3. Visual Studio .NET 学院版

Visual Studio.NET 学院版拥有专业版的所有功能,再加上供教师和学生使用的学院版特定的文档和代码示例,可以让教师和学生快速地掌握使用 Visual Studio.NET 开发应用程序的方法。

1.1.4 Visual Basic 的新特性

一直以来,大家总是针对 Visual Basic 究竟是“面向对象”、“基于对象”还是“面向组件”的语言而争论不休。但是,这些争论在 Visual Basic.NET 面前显得毫无力量,因为,不管依照怎样的标准来定义“面向对象”,Visual Basic.NET 绝对符合面向对象的每一个标准。下面就来看看它具有怎样的新特性。

1. 面向对象的程序设计

传统的结构化的程序设计(数据与程序式的代码分别存储)存在一些缺点。由于数据元素可以从任何代码中访问,因此在没有与开发相关知识的情况下,修改数据是不可能的。这样做可能会导致非常难调试的运行错误。此外,维护可能会成为一项困难的任务;并且,依靠程序员来管理代码和数据将导致非常低的重用率。

面向对象的程序设计(Object-Oriented Programming, OOP)解决了上述这些问题。它将数据以及在数据上采取行动的方法打包在一个称为对象的单位中。一个对象的数据可以被隐藏,以避免出现未授权的修改。此外,这个对象提供了一组公共方法对该数据进行操作,这种概念被称为封装。由于实现细节与接口分离,底层的程序设计逻辑可以在日后修改,因此不用担心破坏调用这个对象的代码。

OOP 还允许开发人员通过继承来重用代码和数据。通过继承预定义的对象,开发人员可以更迅速地构造更复杂的应用程序。由于编写新代码通常有可能增加错误,因此重用代码可以使出现额外错误的几率降到最低。

为了解决这些需求,Visual Basic.NET 将提供这些额外的语言特性,这将使得它成为第一流的具有上述所有好处的、面向对象的程序设计语言。

2. 继承

很久以来,Visual Basic 程序员最迫切的要求是让 Visual Basic 支持继承(Implementation

Inheritance)。Internet 时代的程序开发需要快速的编译和大量的重用，而 Visual Basic 现在包括了可视化窗体继承在内的全部继承的实现，因此，开发人员可以使用新的关键字 Inherits 来从一个已有的类派生。

Inherits 语句支持与继承相关的所有可视化属性。派生类的实例支持这个基类所支持的所有方法和接口。当然，派生类可以扩展基类所支持的方法和接口集合。

派生类可以使用关键字 Overrides 重载基类中定义的方法。为了减少程序设计错误，Visual Basic 可以防止意外地重载一个函数；只有被声明为“Overridable(可重载)”的函数才能在派生类中被重载。

3. 重载

Visual Basic 现在允许进行函数重载，这使得开发人员可以创建有相同名称，但却有不同参数类型、不同版本的 Sub 或 Function。

当对象模型规定，对于操作不同数据类型的过程使用类似的名称时，重载尤其有用。例如，一个可以显示几种数据类型的类可能会有如下的 Display 过程：

```
Overloads Sub Display (theChar As Char)
```

```
Overloads Sub Display (theInteger As Integer)
```

```
Overloads Sub Display (theDouble As Double)
```

如果没有重载，将不得不为每个过程使用截然不同的名称，或者使用一个 Object 参数。重载提供了一个处理多种数据类型的更明确、更有效的方法。

4. 带参数的构造函数

带参数的构造函数(或简单地称为构造函数)允许在创建类的实例的同时将变量传递给新的实例。构造函数对于面向对象的程序设计至关重要，因为它们允许实例的创造者向用户定义的构造代码传递参数。构造函数通过允许在单个表达式中创建和初始化一个新的实例，从而简化了客户代码。

5. 自由线程

当开发人员在 Visual Basic 中创建应用程序时，他们编写的是同步运行的代码。这意味着，每行代码必须在下一行代码之前执行。当开发 Web 应用程序时，可伸缩性是关键。开发人员需要支持并发处理的工具。

通过包含自由线程，开发人员可以产生一个执行某些长期任务、执行复杂的查询或运行多部分计算的线程，而应用程序的其他部分则继续执行，从而提供了异步处理功能。

6. 结构化的异常处理

开发企业应用程序需要构造可重用的、可维护的组件。在以前版本的 Visual Basic 中，Basic 语言的一个具有挑战性的方面是支持错误处理。开发人员发现，一个一致的错误处理模式意味着可以大量地复制代码，而使用原有的 On Error GoTo 语句进行错误处理则使大型应用程序的开发和维护速度变得很慢。它的名字就反映了某些问题：正如 GoTo 所暗示的，当一个错误发生时，控制被转移给子程序中某个标记的位置。一旦错误代码运行，通常它必须通过标准的 GoTo 语句来跳转到错误处理过程，且最终应用程序将使用另一个 GoTo 或

Exit 来退出这个过程。利用各种 Resume 和 Next 组合快速地处理几个不同的错误，将产生难读的代码，并且在执行路径没有完全考虑到时将会导致出现频繁的错误。

利用 Try...Catch...Finally，这些问题迎刃而解，开发人员可以嵌套使用异常处理过程，并且有一个控制结构用于编写在正常和异常条件下都执行的清除代码。

7. 严格的类型检查

以往，Visual Basic 语言在其产生的隐含类型强制转换方面非常自由。现在，通过添加一个新的编译选项，Visual Basic 可以对于可能会在运行时引起错误的任何强制转换生成编译错误。设置 Option Strict On 将会在试图执行不是用户所预期的自动类型转换时产生编译错误，从而提高了类型的安全性。

8. 共享成员

共享成员是指可以被类的所有实例共享的类的数据和函数成员。让一个类的所有实例共享一个数据成员或函数成员的实例，是在使用继承的 Visual Basic 应用程序中所必需的。一个共享的数据成员独立于这个类的任何一个具体的实例而存在。一个共享的方法与普通的方法不同，它并不被隐含地传递类的实例。由于这个原因，在一个共享的方法中，不允许不合格地引用非共享的数据成员。公共的共享成员可以随意访问，并且可以从这个类的某个实例中对它们进行晚期绑定。

9. 初始化工具

Visual Basic.NET 支持在变量被声明的一行中对其进行初始化。初始化可以用在任何位置。例如下面的两组语句意义相同：

```
Dim X As Integer = 1
```

和

```
Dim X As Integer  
X = 1
```

总之，Visual Basic.NET 已经成为第一流的面向对象的程序设计语言。使用 Visual Basic.NET，开发人员将能够创建高度可伸缩的、具有明确的自由线程(Free Threading)代码，并且可以在代码中使用现代化的语言结构，如结构化的异常处理(Structured Exception Handling)，具有很高的可维护性。Visual Basic 将提供给开发人员创建健壮的、可伸缩的分布式 Web 应用程序所需的所有特性。

1.2 Visual Basic.NET 的安装和启动

1.2.1 安装 Visual Basic.NET 的软、硬件要求

作为运行在 .NET 框架上的一门语言，要使 Visual Basic.NET 正常运行，必须先安装好 .NET 运行环境。该运行环境可以从微软网站上下载，网址是：

<http://download.microsoft.com/download/7/b/9/7b90644d-1af0-42b9-b76d-a2770319a568/dotnetfx.exe>

如果系统中没有此运行环境,则在安装 Visual Basic.NET 时,安装程序会提示更新组件,通过组件更新操作,.NET 运行环境就安装到了计算机上。

这里要提到的是,Visual Basic.NET 是 Visual Studio.NET 中的一员,要安装 Visual Basic.NET,就要安装 Visual Studio.NET。(注意:微软最近发布了 Visual Basic.NET 标准版,这个版的 Visual Basic.NET 是可以单独安装的。)

事实上,目前 Visual Studio.NET 有两个版本:一个是 Visual Studio.NET 2002,另一个是 Visual Studio.NET 2003。后一个版本运用更广泛一些。

下面是 Visual Studio.NET 2003 对计算机的软、硬件要求:

- (1) 处理器:具有 Pentium II 450 MHz 或更高处理器的计算机。
- (2) 内存: Windows NT Workstation 4.0 需要 64 MB; Windows 2000 Professional 需要 96 MB; Windows 2000 Server 需要 192 MB; Windows XP Professional 需要 192 MB。
- (3) 硬盘: 2500 MB 硬盘空间,系统驱动器至少需 500 MB 的空间。
- (4) 驱动器: CD-ROM 或 DVD ROM。
- (5) 显示器: Super VGA (800×600) 或 256 色的更高分辨率显示器。
- (6) 操作系统: Windows NT 4.0、Windows 2000、Windows XP 和 Windows Server 2003。
- (7) 鼠标: Microsoft 鼠标或兼容的定点设备。

1.2.2 Visual Basic.NET 集成开发环境(IDE)的安装

上面已经提到过,要运行 Visual Basic.NET,只需要在计算机上安装 .NET 运行环境。而要用 Visual Basic.NET 快速开发应用程序,就要安装 Visual Basic.NET 集成开发环境——Visual Studio.NET。

首先应关闭所有打开的应用程序,以免增加计算机在安装期间重新启动的次数。然后按以下步骤进行操作:

(1) 在光驱中插入标有“Visual Studio.NET 光盘 1”的光盘。

(2) “自动运行”功能将启动 Setup.exe。如果禁用了“自动运行”功能,则从安装 CD 的根目录下运行 Setup.exe。如图 1-1 所示。

(3) 安装程序将扫描磁盘,查找已安装的组件。如果扫描过程中确定系统需要组件更新,则“Visual Studio.NET 安装程序”对话框中会提供“1 Visual Studio.NET 系统必备”,选择第 1 步即可更新系统组件。要安装系统必备,安装程序会提示用户插入系统必备光盘。通常这一步骤安装程序所做的是安装 .NET 运行环境,更新数据访问组件(MDAC),更新 Internet Explorer 等。

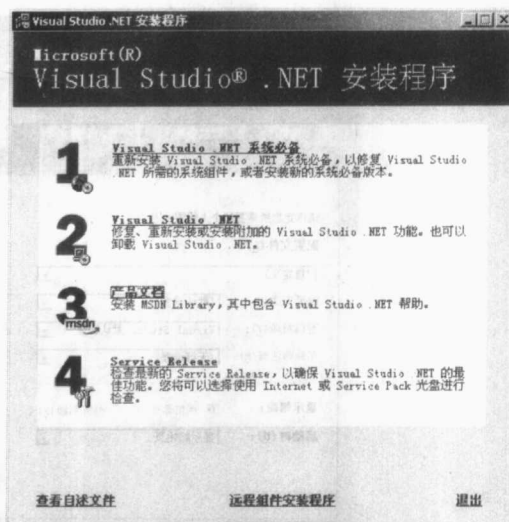


图 1-1 Visual Studio.NET 安装程序主界面