

VB.NET

数据库编程

学习捷径

杨 毅 郭聪宾 刘晓宏 编著



北京科海电子出版社



VB.NET 数据库编程学习捷径

杨 毅 郭聪宾 刘晓宏 编著



北京科海电子出版社



内 容 提 要

VB.NET 是目前用于开发数据库应用程序的首选语言之一。本书从基础知识入手, 结合实例讲解编程方法和技巧, 为读者提供了学习 VB.NET 数据库编程的捷径。

本书共分 12 章, 首先介绍 VB.NET 中的几种访问数据库的方法, 包括 DAO 技术、ADO.NET 技术以及自定义访问数据库的类, 其中穿插了应用程序的界面技术、数据库基本管理、数据库查询等; 然后通过实例介绍了 C/S 和 B/S 两种数据库体系; 最后介绍了应用程序的打包和帮助。书中所附的大量数据库编程实例不仅适合于自学, 稍加改动后更可投入实际应用中。光盘中包含全部实例源代码和可执行文件, 便于读者学习。

本书内容新颖、实用, 叙述浅显易懂, 可以作为大专院校学生的学习用书和培训班的培训教程, 亦可供数据库管理员、系统管理员参考。

品 名: VB.NET 数据库编程学习捷径
作 者: 杨 毅 郭聪宾 刘晓宏
责任编辑: 安 靖
排 版: 卞雨桂
光盘制作: 马首鳌
咨询电话: (010) 82896449/50-8407



出 品: 北京科海电子出版社
印 刷 者: 北京市耀华印刷有限公司
发 行 者: 新华书店总店北京发行所
开 本: 787×1092 1/16 印张: 21.25 字数: 518 千字
版 次: 2003 年 6 月第 1 版 2003 年 6 月第 1 次印刷
印 数: 0001~5000
盘 号: ISBN 7-900107-95-9
定 价: 29.00 元 (1CD/配套手册)

前 言

Microsoft公司总是引导着计算机技术发展的潮流。继Microsoft Visual Studio 6.0之后，Microsoft公司又推出了划时代的Visual Studio .NET。从此，我们迈进了.NET时代。

除了引入了全新的C#以外，.NET在语言上的最大变化来自于对Visual Basic的改进。崭新的Visual Basic.NET是.NET家族中的杰出代表。经过一番洗心革面，Visual Basic焕发了青春。Visual Basic.NET不但继承了Visual Basic一贯的优良传统，比如简单易学、程序简洁，而且抛弃了传统的Visual Basic的很多缺点，还引入了一些堪称是革命性变革的新特点，如可以实现类的完全继承、函数和过程的重载等。

在保持了Visual Basic语言风格的基础之上，Visual Basic.NET广泛吸收了.NET家族中其他语言的优点和长处。这使得其在编译之后（Visual Basic.NET不再有解释执行的方式）产生的代码的执行效率大大提高。所以，无论是作为编程初学者的入门语言还是编程高手们的强大开发工具，Visual Basic.NET都是当之无愧的。

对于本书所要研究的数据库开发领域，Visual Basic.NET无疑更将成为未来开发各种数据库应用程序的首选语言。这是因为数据库开发本来就是Visual Basic的传统强项。根据统计，开发功能相当的数据库时，使用Visual C++所需要的时间大概是使用Visual Basic所需时间的3倍。而且在.NET平台中，Visual Basic.NET可以很容易地结合ADO技术的继任者ADO.NET，而ADO.NET则是未来数据库应用程序开发所使用的核心技术。

本书涵盖了用Visual Basic.NET进行数据库开发的全部内容，详尽介绍了与之相关的理论和实践。本书对各个主题的讲解都与具体的实例相结合，使读者在掌握Visual Basic.NET的语言特点和编程技巧的同时能够很快及时实践，把所学到的知识应用到自己的程序中去。读者细读全书之后，很快就能利用Visual Basic.NET开发出强大的数据库应用程序。

本书所有实例都已在Visual Basic.NET正式版本上调试通过，并且在配书光盘中附有全部实例的源代码和可执行文件，可供读者在开发自己的应用程序时查看。希望读者能对本书所给的程序进行改进，把书中的知识转化为自己的技能。

由于时间仓促，水平有限，书中难免有不妥之处，恳请各位专家、读者批评指正。

作 者

2003.6

目 录

第1章 基础知识	1
1.1 Visual Basic.NET简介	1
1.1.1 Visual Studio.NET的安装	1
1.1.2 Visual Basic.NET的集成开发环境	2
1.1.3 Visual Basic.NET的新特点	7
1.2 数据库基础知识	14
1.2.1 数据库的基本概念	14
1.2.2 关系数据库	15
1.2.3 数据库开发的基本步骤	17
1.3 本章小结	18
第2章 应用程序基本界面技术	19
2.1 Visual Basic.NET的基本控件	19
2.1.1 Button控件	19
2.1.2 ComboBox控件	20
2.1.3 PictureBox控件	20
2.2 Visual Basic.NET的高级控件	22
2.2.1 RichTextBox控件	22
2.2.2 CommonDialog控件	24
2.3 Visual Basic.NET的菜单设计	27
2.3.1 菜单控件的特点	27
2.3.2 菜单设计示例	28
2.4 控件的数据绑定	33
2.4.1 控件的数据绑定简介	33
2.4.2 控件的数据绑定示例	34
2.5 数据窗体向导	37
2.6 本章小结	40
第3章 应用程序高级界面技术	41
3.1 MSChart控件	41
3.1.1 MSChart控件的添加	41
3.1.2 显示三维图表	42
3.1.3 显示数据	44
3.1.4 MSChart控件与数据库的绑定	45

3.2 Repeater控件	46
3.2.1 Repeater控件的添加	46
3.2.2 为Repeater控件定义模板	47
3.2.3 Repeater控件与数据库的绑定	49
3.2.4 运行结果	49
3.3 TreeView控件和ListView控件的组合	49
3.3.1 资源管理器式界面简介	50
3.3.2 资源管理器式界面的实现	50
3.4 本章小结	60
第4章 基本数据库管理与查询	61
4.1 SQL简介	61
4.2 自制数据库管理工具	61
4.2.1 界面设计	61
4.2.2 代码实现	62
4.3 表管理	65
4.3.1 表的创建	65
4.3.2 表的删除	67
4.3.3 表的更新	68
4.4 记录管理	69
4.4.1 数据的添加	69
4.4.2 数据的修改	70
4.4.3 数据的删除	71
4.5 索引管理	71
4.5.1 索引的建立	71
4.5.2 索引的删除	71
4.6 视图管理	71
4.6.1 视图的创建	71
4.6.2 视图的删除	72
4.6.3 视图的更新	73
4.7 基本查询	73
4.7.1 使用SELECT...FROM检索记录	73
4.7.2 使用WHERE说明条件	74
4.7.3 使用GROUP BY对查询结果分组	76
4.7.4 使用HAVING子句	76
4.7.5 使用ORDER BY对查询结果排序	76
4.7.6 集合函数	77
4.8 高级查询	77
4.8.1 多重表的查询	77

4.8.2 子查询.....	79
4.8.3 功能查询.....	82
4.9 用Visual Basic.NET实现数据库查询.....	83
4.9.1 使用Visual Basic.NET的查询工具.....	83
4.9.2 编写数据库查询工具程序.....	89
4.10 本章小结.....	91
第5章 ADO.NET访问数据库.....	92
5.1 ADO.NET中的重要概念与对象.....	92
5.1.1 无连接的编程模型: Managed Provider.....	92
5.1.2 内存中的数据库: DataSet.....	93
5.2 ADO.NET访问数据库的方法.....	94
5.2.1 创建DSN.....	94
5.2.2 ADO.NET与数据库的连接.....	95
5.2.3 测试与数据库连接示例.....	95
5.2.4 DataSet对象的获得.....	98
5.2.5 DataTable对象的操作.....	101
5.2.6 DataSet对象的显示.....	105
5.2.7 动态创建DataSet对象并显示的示例.....	105
5.2.8 OleDbCommand对象的使用.....	108
5.2.9 OleDbDataReader对象.....	110
5.2.10 OleDbDataReader对象使用示例.....	111
5.2.11 数据控件的数据绑定.....	113
5.3 本章小结.....	113
第6章 用类访问.....	114
6.1 类和对象的使用.....	114
6.1.1 “类”与“对象”的概念.....	114
6.1.2 类的继承.....	115
6.1.3 实现类与对象的示例.....	116
6.2 使用类访问数据库.....	122
6.2.1 单个记录处理类的规划.....	122
6.2.2 建立测试类的工程.....	123
6.2.3 单个记录处理类的代码的编写.....	123
6.2.4 测试工程的界面设计.....	128
6.2.5 测试工程的代码设计.....	131
6.3 数据输出类的创建.....	135
6.4 本章小结.....	139

第7章 计算机选课系统	140
7.1 系统的设计	140
7.1.1 系统的需求分析	140
7.1.2 系统的概念结构设计	140
7.2 数据库的实现	141
7.2.1 使用Access建立数据库	141
7.2.2 使用数据工具完成数据库	143
7.3 程序实现	147
7.3.1 总体流程分析	147
7.3.2 全局变量的定义	148
7.3.3 与数据库的连接	148
7.3.4 “登录”功能的实现	149
7.3.5 “修改密码”功能的实现	154
7.3.6 “选课”功能的实现	158
7.3.7 “选课信息”功能的实现	167
7.4 本章小结	169
第8章 客户端/服务器程序的创建	170
8.1 客户端/服务器体系结构	170
8.1.1 客户端/服务器简介	170
8.1.2 客户端/服务器的结构	171
8.2 客户端/服务器体系结构的实现	174
8.2.1 采用客户端/服务器体系结构的理由	174
8.2.2 数据源的访问接口技术	175
8.3 本章小结	177
第9章 人力资源管理系统	178
9.1 系统的设计	178
9.2 用Access实现数据库	179
9.3 新建工程	180
9.4 闪屏窗体的设计	181
9.4.1 闪屏窗体的界面设计	181
9.4.2 闪屏窗体的代码设计	182
9.5 登录窗体的设计	183
9.6 主窗体中的高级控件	185
9.6.1 ToolBar控件的使用	185
9.6.2 ImageList控件的使用	186
9.6.3 DataGrid控件的使用	187
9.6.4 TabControl控件的使用	187
9.6.5 GroupBox控件的使用	188

9.6.6	MonthCalendar控件的使用	188
9.6.7	MaskedTextBox控件的使用	189
9.7	主窗体的设计	192
9.7.1	初始公共界面	193
9.7.2	登录过程	196
9.7.3	3个基本的页面与3个基本公用过程	201
9.7.4	数据库管理员登录后的界面	205
9.7.5	总经理登录后的界面	217
9.7.6	经理登录后的界面	220
9.7.7	职员登录后的界面	222
9.8	本章小结	224
第10章	浏览器/服务器程序的创建	225
10.1	浏览器/服务器体系结构	225
10.1.1	浏览器/服务器体系结构的优势	225
10.1.2	浏览器/服务器模式的工作流程	226
10.1.3	浏览器/服务器的3种工作方式	227
10.2	Web数据库开发	228
10.2.1	Web数据库简介	228
10.2.2	Web数据库技术	228
10.3	ASP与ASP.NET	229
10.3.1	HTML简介	229
10.3.2	ASP简介	234
10.3.3	ASP.NET简介	240
10.4	本章小结	244
第11章	图书馆网上借阅系统	245
11.1	系统设计	245
11.1.1	系统的需求分析	245
11.1.2	系统的概念结构的设计	245
11.2	用Access实现数据库	246
11.3	借阅流程分析	251
11.4	系统的具体实现	252
11.4.1	创建工程	252
11.4.2	登录页面的设计	253
11.4.3	书籍查询页面的设计	266
11.4.4	书籍借阅页面的设计	285
11.4.5	查看个人借阅记录的页面的设计	295
11.4.6	更改个人密码的页面的设计	299
11.5	本章小结	303

第12章 应用程序的帮助系统和打包发行	304
12.1 应用程序的帮助系统	304
12.1.1 帮助文件简介	304
12.1.2 图形编辑的有力工具	305
12.1.3 帮助文件制作工具	308
12.1.4 制作上下文敏感帮助	318
12.2 应用程序的打包发行	320
12.2.1 新建安装工程	320
12.2.2 文件系统编辑器	321
12.2.3 注册到注册表	323
12.2.4 编辑文件类型与文件关联	324
12.2.5 安装程序的界面设计	326
12.2.6 编译工程	328
12.3 本章小结	328

第1章 基础知识

1.1 Visual Basic.NET简介

1.1.1 Visual Studio.NET的安装

Microsoft公司新推出的Visual Studio.NET融合了Visual Basic.NET、Visual C#、Visual C++.NET等诸多语言。作为Visual Studio.NET中的一部分，Visual Basic.NET并没有单独的安装程序，必须通过在安装Visual Studio.NET时选择Visual Basic.NET组件的方式来安装。

系统要求

	最低配置	推荐配置
CPU主频	P II 450MHz	PIII 733MHz
内存空间	128MB	256MB
硬盘空间	3GB	5GB

安装过程

首先插入第1张安装光盘，安装程序将自动引导至如图1-1所示的画面。

如果是第1次安装Visual Studio.NET，那么首先需要对计算机系统进行Windows组件更新，否则安装程序不会继续进行。更新的内容包括：

- Windows 2000操作系统要安装Windows Service Pack 2。
- Windows NT要安装Windows Service Pack 6。
- 可以选择安装IE 6.0。

完成Windows组件更新后，安装程序将会自动地进行必要的系统更新，安装如图1-2中所示的几个软件包。

系统更新完成之后，就可以安装Visual Studio.NET了。如图1-3所示。

在左侧的树状列表中选择要安装的内容，同时还可以选择安装的路径（安装程序统计出了各个硬盘空间的使用情况，供选择路径时参考）。

当选择完安装的内容和路径之后，下面的工作就是按照提示，依次插入光盘进行安装。

安装完成

安装完成后，单击【Done】按钮结束安装过程。

此时在Windows中单击【开始】|【程序】，可以发现【程序】菜单中多了3个新的项目，分别是Microsoft Visual Studio.NET、HTML Help Workshop和Microsoft.NET Framework SDK。

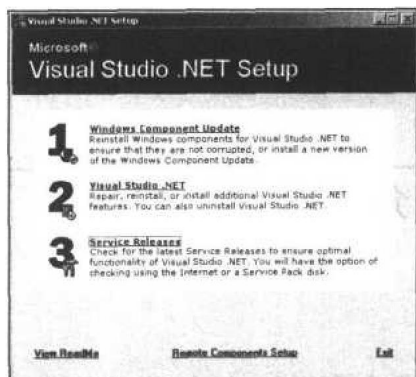


图 1-1 安装引导界面

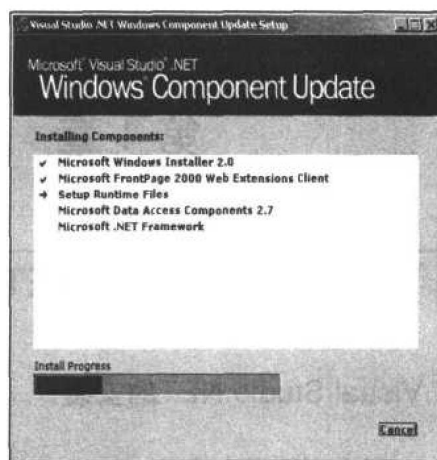


图 1-2 系统更新界面

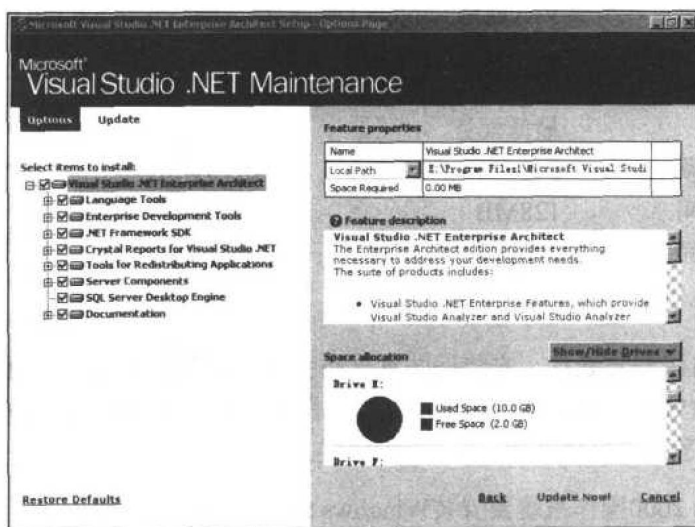


图 1-3 选择安装内容和路径

1.1.2 Visual Basic.NET的集成开发环境

集成开发环境，即IDE（Integrated Development Environment），是一系列开发工具的集成，它为软件开发人员提供了全方位的服务。

作为Visual Studio.NET的一部分，Visual Basic.NET的集成开发环境也是被融合在Visual Studio.NET的集成开发环境之中的。

Visual Studio.NET 的集成开发环境

启动Visual Studio.NET的步骤为：在Windows中单击【开始】|【程序】|【Microsoft Visual Studio.NET】。

Visual Studio.NET的集成开发环境如图1-4所示。

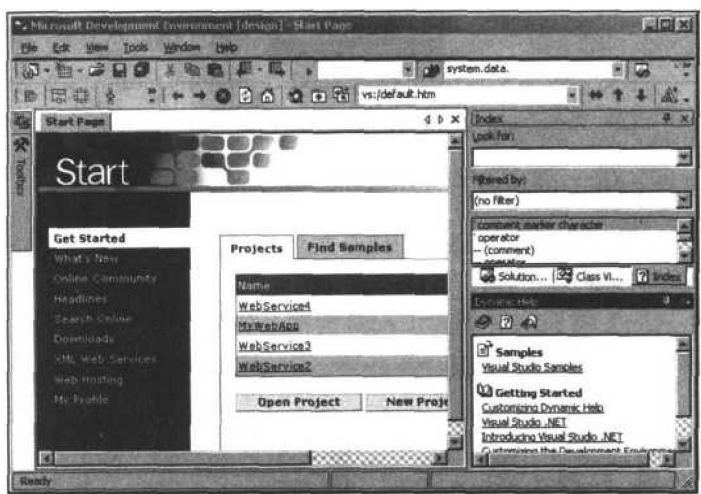


图 1-4 Visual Studio.NET 的集成开发环境

在这个全新的集成开发环境中，Microsoft公司采用了很多的新技术。可以明显地感觉到，这个集成开发环境很像是一个上网用的浏览器。在图1-4中，可以看到一个【Start Page】页面。它是默认的开始页面。

在【Start】页面中，共有两个标签页面：【Projects】和【Find Samples】。

- 【Projects】页面可以很方便地管理所编写的各种工程。
- 【Find Samples】页面可以找到很多已经开发出来的丰富而详实的实例。通过学习这些实例可以极大地加快学习.NET技术的进程，更好地掌握.NET技术的精髓。单击【Find Samples】后的界面如图1-5所示。

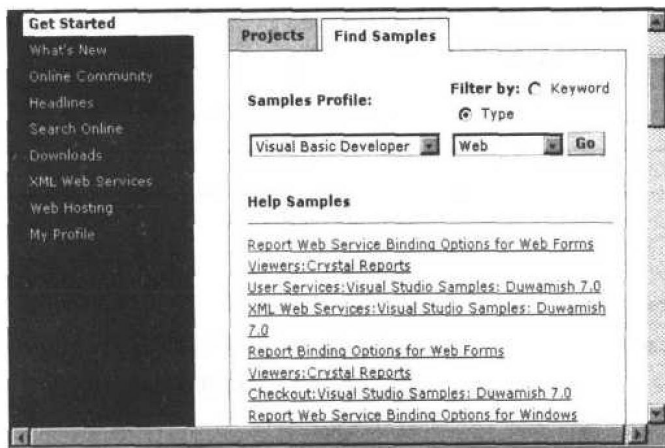


图 1-5 查看例子

它可以通过使用Keyword或Type作为查看例子的过滤器来缩小查找的范围。当用Keyword过滤器时，可以在其中的对话框中输入要查找的例子名称；用Type过滤器时，可以选择3种类型。如图1-6所示。

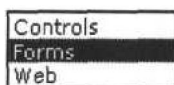


图 1-6 用 Type 时可选择的类型

进入 Visual Basic.NET 的集成开发环境

上面介绍的基本上是 Visual Studio.NET 开发环境的总体特点，下面将具体介绍 Visual Basic.NET 开发环境的特点。

单击【New Project】按钮打开新建工程对话框，如图1-7所示。

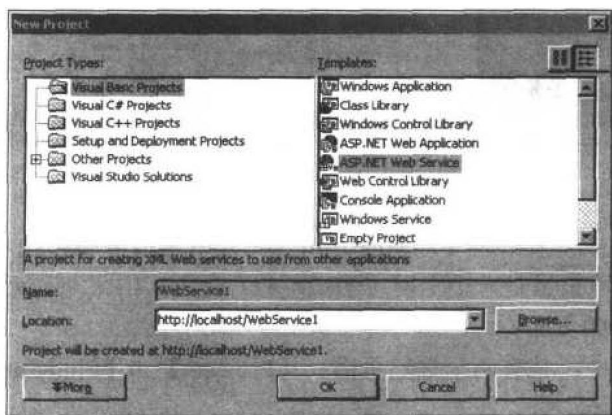


图 1-7 新建工程对话框

其中，【Project Types】中列出了 Visual Studio.NET 可以建立的工程种类，包括【Visual Basic Projects】、【Visual C# Projects】、【Visual C++ Projects】等。要建立 Visual Basic.NET 工程，只需选中【Visual Basic Projects】即可，这时，在【Templates】中列出可供其建立的程序文件类型，各种文件类型的名称与意义见表1-1。

表1-1 Visual Basic.NET可建立的工程类型及其意义

工程类型名称	意义
Windows Application	建立窗口应用程序
Class Library	建立类库
Windows Control Library	建立窗口控件类库
ASP.NET Web Application	建立ASP.NET网络应用程序
ASP.NET Web Service	建立ASP.NET网络服务器端程序
Web Control Library	建立网络控件库
Console Application	建立控制台程序
Windows Service	建立窗口服务器程序
Empty Project	建立一个空的工程
Empty Web Project	建立一个空的网络工程
New Project In Existing Folder	在已经存在的文件夹中建立新的工程

如果要建立一个窗口应用程序，先选中【Windows Application】；然后在【Name】文本框中输入工程的名称，必要时在【Location】文本框中修改文件的路径；最后单击【OK】按钮就可以了。Visual Basic.NET的集成开发环境如图1-8所示。

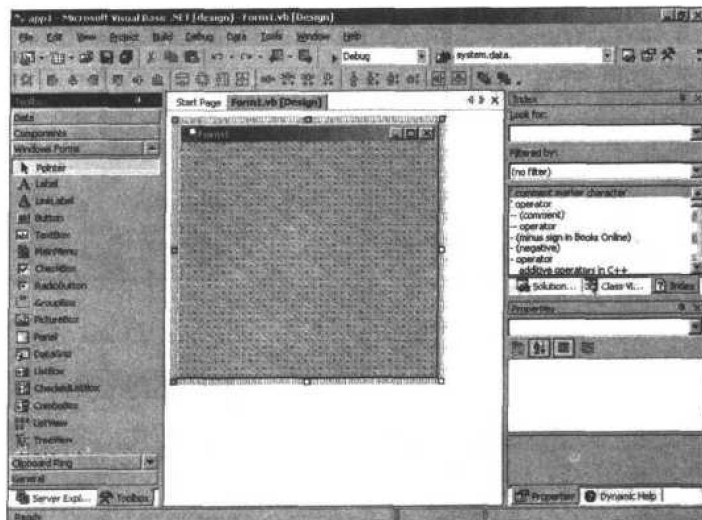


图 1-8 Visual Basic.NET 的集成开发环境

界面的上方是菜单栏和工具条，包含了一些常用的命令。

左下方是【Server Explorer】面板和【Toolbox】面板。【Server Explorer】面板中列出了当前连接的服务器，如图1-9所示。【Toolbox】面板中列出了窗体设计所需的基本控件，如图1-10所示。

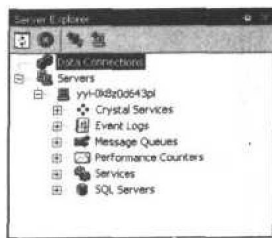


图 1-9 【Server Explorer】面板



图 1-10 【Toolbox】面板

中间部分是对窗体进行可视化设计、编辑代码及浏览在线帮助的区域。可通过单击上面的标签进行视图切换。

右方的浮动面板包括【Solution Explorer】面板、【Class View】面板、【Dynamic Help】面板和【Properties】面板等，分别如图1-11至图1-14所示。这些面板可通过菜单【View】

进行添加和删除。



图 1-11 【Solution Explorer】面板

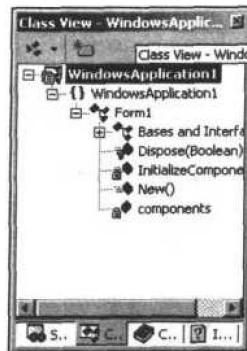


图 1-12 【Class View】面板

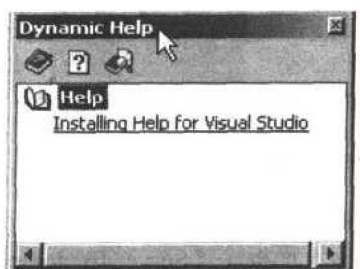


图 1-13 【Dynamic Help】面板



图 1-14 【Properties】面板

值得一提的是，在Visual Studio.NET中加入了联机帮助功能。可以通过在菜单中单击【Help】|【Index】激活帮助面板。如图1-15所示。

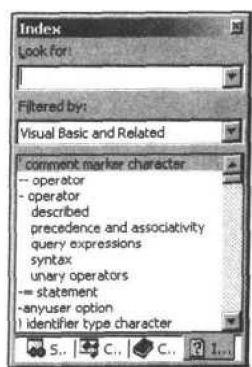


图 1-15 【Index】面板

这个帮助面板的用法如下：在【Look for】文本框中输入要查找的命令、方法或对象等相关的单词；在【Filtered by】文本框内选择查询的空间，如Visual Basic、Visual Basic and Related、.net Framework SDK、Visual C++、Visual C#等，通过这一选项可以框定要查

找的范围，大大缩小查找时间和工作量。这时，在下面的文本框中就会枚举出所有与要查找的关键词匹配的对象、方法或命令等的名称。双击显示的名称即可通过设计区的浏览器对联机帮助进行浏览。

1.1.3 Visual Basic.NET的新特点

介绍Visual Basic.NET的新特点是以Visual Basic 6.0为参照的。Visual Basic.NET融合了Visual Basic 6.0及其以前版本的语言风格，简单易学，又吸收了其他语言（如C++）的优点，作出了种种改进，提高了代码的运行效率，增强了语言的功能。下面就对Visual Basic.NET的新特点进行初步介绍，为以后的编程打好基础。

语言方面的改进

声明的改进

Visual Basic.NET的声明的语法为：

```
[ <attrlist> ] [ { Public | Protected | Friend | Protected Friend | Private
    | Static } ] [ Shared ] [ Shadows ] [ ReadOnly ] Dim [ WithEvents ]
name [ (boundlist) ] [ As [ New ] type ] [ = initexpr ]
```

与Visual Basic 6.0相比，它有如下明显变化：

（1）初始化的改进

在Visual Basic.NET中，可以在声明变量的同时对变量进行初始化。例如：

```
Dim str As String = "Visual Basic.NET"
```

这相当于Visual Basic 6.0中的如下代码：

```
Dim str As String
str = "Visual Basic.NET"
```

（2）只读变量的引入

只读变量不像常数，声明它的时候不用初始化，但是一旦给它赋值，就不能再修改。

请看下面的例子：

```
ReadOnly Dim a As Float
'a为只读型变量
a=123.4556
'a初始化
a=3.14159
'a再次赋值，出错
```

（3）同类型变量的声明

看下面这一语句：

```
Dim a, b As Boolean
```

在Visual Basic.NET中，a与b均被声明为Boolean型的变量；在Visual Basic 6.0中，只有b