全国高职高专规划教材

Visual Basic .NET 程序设计教程

Programming Visual Basic .NET

吴小松 主 编
王 文 副主编

全国高职高专规划教材

科学出版社
www.sciencep.com

全国高职高专规划教材

Visual Basic .NET 程序设计教程

吴小松　主　编

王　文　副主编

科学出版社

北　京

内 容 简 介

本书详细地介绍了 Visual Basic .NET 中文版的编程环境、面向对象的程序设计基础知识、程序的控制结构、数组、过程、常用内部控件的使用、菜单栏、工具栏、对话框、数据库和网络程序设计技术。

通过对本书的学习，读者可以掌握 Visual Basic .NET 程序设计的基本知识、软件界面的设计、数据库程序的设计及网络编程的方法。本书概念清晰、逻辑性强、层次分明。本书既可作为高职、高专及各类中等学校及社会培训班的教材，也可供初学编程的读者自学使用。

图书在版编目(CIP)数据

Visual Basic.NET 程序设计教程/吴小松主编. —北京：科学出版社，2004

（全国高职高专规划教材）

ISBN 7-03-012776-5

Ⅰ.V… Ⅱ.吴… Ⅲ.BASIC 语言—程序设计—高等学校：技术学校—教材 Ⅳ.TP312

中国版本图书馆 CIP 数据核字(2004)第 002759 号

策划编辑 李振格/责任编辑 王日臣

责任印制 吕春珉/封面设计 东方人华平面设计部

科学出版社 出版

北京东黄城根北街16号

邮政编码 100717

http://www.sciencep.com

双青印刷厂 印刷

科学出版社发行 各地新华书店经销

2004 年 2 月第 一 版 开本：787×1092 1/16

2004 年 2 月第一次印刷 印张：13

印数：1—5 000 字数：288 000

定价：18.00 元

（如有印装质量问题，我社负责调换〈双青〉）

全国高职高专规划教材编委会名单

主　任：俞瑞钊

副主任：陈庆章　周必水　刘加海

委　员：（以姓氏笔画为序）

方　程　方锦明　王筱慧　王　雷　代绍庆　卢菊洪

刘向荣　吕何新　孙光弟　朱　炜　江爱民　江锦祥

余根墀　张　元　张学辉　张锦祥　张德发　李天真

李永平　李良财　李明钧　李益明　汪志达　沈凤池

沈安衢　邵应珍　陈月波　陈晓燕　周国民　周建阳

范剑波　胡海影　赵小明　凌　彦　徐文杰　欧阳江林

秦学礼　戚海燕　曹哲新　章剑林　龚祥国　董方武

蒋黎红　谢　川　谢晓飞　鲁俊生　楼　丰　楼程伟

鞠洪尧

秘书长：熊盛新

本书编写人员名单

主　编：吴小松

副主编：王　文

撰稿人：吴小松　王　文　沈　杰　姜柏军　马玉山

前　言

随着计算机的迅速普及，网络应用的“百姓化”，网络已经成为构建新一代操作系统的基础。因此，微软公司在以.NET 作为发展的核心策略指导下，重新编写了 Visual Studio .NET 系列产品。Visual Basic .NET 是其中一个重要的组成部分，它支持许多新的面向对象语言的特性，并增添了一些新的数据类型以及结构化错误控制等，与以前版本相比，可以说是脱胎换骨。

本书共分 12 章，第 1 章主要介绍 Visual Basic .NET 的基础知识；第 2 章主要是初步介绍使用 Visual Basic .NET 开发 Windows 程序的基本知识；第 3 章主要介绍 Visual Basic .NET 程序设计中变量和常量的使用方法、数据类型的基本种类、数组的使用方法以及运算符的使用方法；第 4 章主要介绍条件判断语句、循环结构语句及条件转移语句的使用方法；第 5 章主要介绍事件过程和函数过程；第 6 章主要介绍了面向对象程序设计的基本概念；第 7 章主要介绍 Windows 标准窗体的制作及常用控件的使用；第 8 章主要介绍应用程序对话框的设计；第 9 章主要介绍 Web 技术的使用方法；第 10 章主要介绍关系数据库的使用；第 11 章主要介绍 ADO.NET 对数据进行访问的使用方法；第 12 章主要介绍程序的调试及出错的处理。

本书由吴小松担任主编，负责编写第 3 章、第 5 章、第 6 章和第 7 章；王文担任副主编，负责编写第 9 章和第 12 章；沈杰参编，负责编写第 1 章和第 2 章；姜柏军参编，负责编写第 8 章、第 10 章和第 11 章；中国建设银行辽宁省分行科技处马玉山（工程师、网络科科长）参编，负责编写第 4 章。

在编写过程中，我们力求语言简洁明了，把自己在工作过程的经验溶入教材，由阚晓初老师任主编的实训教材与本书配套。但是由于时间仓促，难免有错，敬请广大读者和同行批评指正。

本书的编写全过程得到沈凤池老师的大力支持和精心指点，在此表示衷心感谢。

编　者

2003 年 10 月

目　录

第1章 Visual Basic .NET简介

本章提要

Visual Basic .NET是美国Microsoft公司推出的重量级软件开发平台Visual Studio .NET的重要成员之一，也是Visual Basic 6.0的升级版。有了Visual Basic .NET，作为创建运行于Microsoft Windows操作系统上的应用程序的最有效工具，Visual Basic的传统地位得以继续保持。Visual Basic开发人员现在可以利用其知识和技能来创建下一代的Web应用程序和XML Web服务。

通过本章的学习，认识.NET为何物，并学会Visual Studio .NET开发平台环境的使用；了解和掌握Visual Basic .NET语言的新特性。

1.1 了解.NET

微软公司提出的.NET 概念，正从各个方面渗入到我们的生活当中。它的使用诚如一位业内名家所描述的那样："请忘掉你认为你所知道的，.NET将改变一切!"。既然如此，作为已经或将要使用Visual Basic .NET开发程序的程序员来说，快速领会这个新概念的含义就显得非常重要了。本节将介绍有关.NET的概念方面的内容。

1.1.1 何谓.NET

究竟什么是.NET呢？微软对此的解释是：.NET是Microsoft XML Web services平台。XML Web services允许应用程序通过Internet进行通讯和共享数据，而不管所采用的是哪种操作系统、设备或编程语言。.NET平台提供创建XML Web services并将这些服务集成在一起。.NET 也是一个用户环境，是一组基本的用户服务，可以作用于客户端、服务器以及其他任何地方，并与该编程模式具有良好的一致性且有新的创意。

1.1.2 .NET对IT专业人员的意义

.NET不仅仅是一个用户体验，更是开发人员体验的集合，且对IT专业人员有着重要意义。目前，IT专业人员能够利用与构建.NET平台相同的技术。.NET Enterprise Servers和Windows 2000操作系统，为创建具有高度可管理性的、能迅速投入市场的应用程序奠定了坚实基础。它们利用的是可扩展标记语言（XML），因此随着Web体系结构的革新，在此平台上创建的程序依然很有价值。

.NET 平台的核心是采用有效的、分门别类的方式来构建应用程序，达到其前所未有的规模。该平台上的 Web 服务模型指的是企业应用程序的中心业务要素通常由本地管理，而支持它们的服务（如用户认证、文件存储、用户首选项管理、日历、邮件等）却无需本地管理，可以方便地订购。为了存储用户文件和邮件，IT专业人员往往在服务

器上安装新的独立磁盘冗余阵列（RAID 阵列），而有了.NET，他们在这一方面将会花费较少的精力，而更多地致力于怎样为公司增加效益。该 Web 服务模型还将动态配置新软件的发布和更新。用户将以极其紧密的连接方式工作，因此更易于管理。而简化的管理又可使 IT 专业人员更能适应变幻莫测的业务需求。

开发应用程序的.NET Web 服务模型将为企业应用程序的创建开辟一条新路。通过企业内外多种服务的联合，很容易把企业内部数据和客户及合作伙伴的相关数据结合在一起，大大简化了应用程序的创建过程。这就为最终用户发掘了空前的功能涵盖性。例如，利用某公司的雇员福利程序，可以从其数据库订购信息，通过 Web 订购福利管理公司的服务、订购工资管理公司的服务。终端用户可以在简单、直观的界面下操作，而这个界面可以显示他们的累积休假时间、个人所得福利以及上次工资额。

1.1.3 了解“软件作为服务”的意义

.NET 的设计目标之一就是为了实现“软件作为服务”。在人们日常生活中，离不开各式各样的服务。例如，用户在当地 ISP 商申请了网络使用帐户，那么当网络设备连接妥当后，就可以享受因特网服务了。其实在现实生活中，我们经常通过某些服务进行“订阅”，并按时缴纳服务费用后，才得以享受相关服务。

比如，像 Word 这样的文字处理软件，虽然其功能强大，但是仍然有很多用户对其感到不满意，因为 Word 中包含了太多根本就不需要的功能。用户都向往着能够对软件进行自由配置，因为只有这样，才可以在得到足够功能的同时，而不必为没用的软件功能白白付费。

随着.NET 应用的普及，软件最终可以实现在因特网上的动态安装，用户可以只安装需要的部分。就好比人们现在使用的基于 Web 的电子邮件系统，只要连接因特网，不论使用的是哪一台计算机，都可以自由地浏览、编辑自己的邮件。这种方式还有助于拉近软件开发者和最终用户的关系，因为一方面，用户可以从频繁的升级和补丁中获得软件的新增功能，而另一方面，软件开发者也可更好地理解用户的需求，进而开发出更受消费者欢迎的软件产品。

1.2 Visual Basic .NET 语言的新特性

Visual Basic 语言有一个很长的更新历史，它反映了 Windows 平台中的根本性的变化。例如，QuickBasic 为支持 Windows 3.0 GUI 开发所作出的重大变化导致了 Visual Basic 的首次发布。在 Visual Basic 4.0 中，向基于 COM 的程序设计转移导致了用于创建 DLL 的语言结构的出现。而在 Visual Basic 5.0 中，语言发展为支持创建 COM 控件。利用每次微软公司的修订，Visual Basic 的普及有了飞速的增长。新的面向对象的语言特性提供给创建企业 Web 应用程序的开发人员的启动功能，将最有可能继续这种趋势。然而，从 Visual Basic 6.0 升级至 Visual Basic .NET 却又是革命性的。针对在以前 Visual Basic 版本语言中对面向对象程序设计支持欠佳的情况，新版的 Visual Basic .NET 中支持包括实现继承在内所有的面向对象的语言特性。利用这些新的语言特性及其另外的一些加强功能，Visual Basic .NET 将迅速有效地提供开发企业关键的应用程序所需的所有强大的

功能，而同时保持使其成为世界上最流行的开发工具的即时可达性。

1.2.1　面向对象编程特性

对于传统的结构化的程序设计（数据与程序式的代码分别存储）存在几个缺点。作为结构化的代码编写的任何代码都不是模块化的。由于数据元素可以从任何代码中访问，因此在没有开发人员的情况下，修改数据是不可能的，这可能会导致非常难调试的运行错误。此外，维护也可能会成为一项实质性的任务。试图理解使用程序式设计改变一行代码的全局影响可能会非常困难。最后，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（OOP，Object-oriented Programming）解决了这些问题。它将数据以及在数据上采取行动的方法打包成一个称为对象的单位中。一个对象的数据可以被隐藏起来，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这个数据进行操作。这种概念称为封装。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，而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重用代码和数据。通过继承预定义的对象，开发人员可以迅速地构造更复杂的应用程序。由于编写新代码有可能加进错误，因此重用代码可以使额外错误出现的机率降到最低。

为了解决这些需求，Visual Basic .NET 将提供这些额外的语言特性，因此被称为第一流的具有上述的所有好处的、面向对象的程序设计语言。

1. 继承

长期以来，开发人员对 Visual Basic 最突出的请求是支持实现继承（implementation inheritance）。Internet 时代的程序开发需要快速的编译和大量的重用。Visual Basic 现在包括了可视化窗体继承在内的全部实现继承。

开发人员可以使用新的关键字 Inherits 从一个已有的类派生。

```
Class1
    Function GetCustomer ()
    ...
    End Function

Class2
    Inherits Class1
    Function GetOrders ()
    ...
    End Function
```

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。

2. 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，但却有不同的自变量类型的、不同版本的 Sub 或 Function。

当你的对象模型规定，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所示的 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，你将不得不为每个过程使用截然不同的名称，或者使用一个 Variant 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。

3. 带参数的构造器

带参数的构造函数（或简单地称为构造函数）允许创建类的新的实例，而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。它们通过允许在单个表达式中创建和初始化一个新的实例来简化客户代码。

1.2.2 其他现代化语言特性

Visual Basic .NET 除比 Visual Basic 6.0 新增了面向对象编程实现外，Visual Basic .NET 还添加了许多额外的结构，简化了更健壮、更具可伸缩性的应用程序的开发。这些特性包括自由线程、结构化的异常处理、严格的类型安全性以及生产力特性，如初始化工具和共享成员。

1. 自由线程

当开发人员在 Visual Basic 中创建应用程序时，他们编写的代码是同步的。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的计算的线程，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。

```
Sub CreateMyThread ()
    Dim b As BackGroundWork
    Dim t As Thread
    Set b = New BackGroundWork ()
    Set t = New Thread
         (New ThreadStart (AddressOf b.Doit) )
        t.Start
End Sub
Class BackGroundWork
```

```
    Sub DoIt ()
    ...
    End Sub
End Class
```

2. 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就是支持错误处理。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。使用现有的 On Error GoTo 语句进行错误处理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 依靠另一个清除位置转移，它最终将使用另一个 GoTo 或 Exit 来推出这个过程。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频繁错误的出现。

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

```
Sub SEH ()
    Try
      Open "TESTFILE" For Output As #1
      Write #1, CustomerInformation
    Catch
      Kill "TESTFILE"
    Finally
      Close #1
    End try
End Sub
```

3. 严格的类型检查

如今，Visual Basic 语言在其产生的隐含类型强制方面非常自由。对于变量以及传递而非引用的参数类来说，Visual Basic 编译器通过产生运行强制，可以将几乎所有的数据类型转换成其他类型。如果要转换的数值不能在没有数据丢失的情况下进行转换，则运行强制操作将实效。通过添加一个新的编译选项，Visual Basic 可以使可能在运行时引起错误的任何强制生成编译错误。选项“Strict”通过在需要进行转换而转换可能会在运行时失败，或像数字类型和字符串之间的自动类型转换在不是用户所预期的时候生成错误，从而提高了类型安全性。

4. 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。在一个类的所有实例间共享一个数据成员或函数成员的实例，在使用继承的 Visual Basic 应用程序中是必需的。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。一个共享的方法是与普通的方法不同，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