

21世纪高等学校计算机规划教材

21st Century University Planned Textbooks of Computer Science

Visual Basic.NET 程序设计教程

Visual Basic.NET Programming

陈志泊 编著

- 实例、习题丰富实用
- 提供所有例题源代码、习题答案
- 紧密结合当代流行开发工具



高校系列

 人民邮电出版社
POSTS & TELECOM PRESS

21世纪高等学校计算机规划教材

21st Century University Planned Textbooks of Computer Science

Visual Basic.NET 程序设计教程

Visual Basic.NET Programming

陈志泊 编著



高校系列

人民邮电出版社

北京

图书在版编目 (CIP) 数据

Visual Basic.NET程序设计教程 / 陈志泊编著. —北京:
人民邮电出版社, 2009. 4
21世纪高等学校计算机规划教材
ISBN 978-7-115-19295-0

I. V… II. 陈… III. BASIC语言—程序设计—高等学校—
教材 IV. TP312

中国版本图书馆CIP数据核字 (2008) 第199738号

内 容 提 要

本书以 Visual Basic 2005 中文版为语言背景, 以 Visual Studio 2005 为开发环境, 结合大量实例, 深入浅出地介绍了 Visual Basic.NET 程序设计语言的主要知识点及程序设计的方法。全书共分 13 章, 主要包括: Visual Basic.NET 程序开发环境、Visual Basic.NET 语言基础、Visual Basic.NET 可视化程序设计初步、Visual Basic.NET 程序控制结构、数组、过程、常用基础控件、菜单与工具栏设计、对话框、Visual Basic.NET 面向对象程序设计、文件、Visual Basic.NET 图形程序设计、Visual Basic.NET 数据库管理应用程序开发等。

本书既重视理论知识的讲解, 又强调应用技能的培养, 每一章都设计了大量的例题对核心知识点进行讲解, 将重要的知识点穿插于具体的实例中。

本书可作为高等学校计算机及其相关专业教材, 也可作为广大计算机爱好者的自学读物。

21 世纪高等学校计算机规划教材

Visual Basic.NET 程序设计教程

-
- ◆ 编 著 陈志泊
责任编辑 滑 玉
执行编辑 刘 博
 - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京艺辉印刷有限公司印刷
 - ◆ 开本: 787×1092 1/16
印张: 21.5
字数: 561 千字 2009 年 4 月第 1 版
印数: 1-3 000 册 2009 年 4 月北京第 1 次印刷

ISBN 978-7-115-19295-0/TP

定价: 34.80 元

读者服务热线: (010)67170985 印装质量热线: (010)67129223
反盗版热线: (010)67171154

编审委员会

主 任：冯博琴

成 员：（按姓氏笔画为序）

毛汉书	王春玲	冯博琴	刘炳文	刘乃琦	陈 明
陈志泊	李盘林	杨 庚	杨小平	张立科	姚 琳
姚彦兵	徐秋红	徐士良	曹永存	黄心渊	蒋本珊

出版者的话

现今社会对人才的基本要求之一就是应用计算机的能力。在高等学校,培养学生应用计算机的能力,主要是通过计算机课程的体制改革,即计算机教学分层、分类规划与实施;密切联系实际,恰当体现与各专业其他课程配合;教学必须以市场需求为导向,目的是培养高素质创新型人才。

人民邮电出版社经过对教学改革新形势充分的调查研究,依据目前比较成熟的教学大纲,组织国内优秀的有丰富教学经验的教师编写一套体现教学改革最新形势的“高校系列计算机教材”。在本套教材的出版过程中,我社多次召开教材研讨会,广泛听取了一线教师的意见,也邀请众多专家对大纲和书稿做了认真的审读与研讨。本套教材具有以下特点。

1. 覆盖面广,突出教改特色

本套教材主要面向普通高等学校(包括计算机专业和非计算机专业),是在经过大量充分的调研基础上开发的计算机系列教材,涉及计算机教育领域中的所有课程(包括专业核心骨干课程与选修课程),适应了目前经济、社会对计算机教育的新要求、新动向,尤其适合于各专业计算机教学改革的特点特色。

2. 注重整体性、系统性

针对各专业的特点,同一门课程规划了组织结构与内容不同的几本教材,以适应不同教学需求,即分别满足不同层次计算机专业与非计算机专业(如工、理、管、文等)的课程安排。同时本套教材注重整体性的策划,在教材内容的选择上避免重叠与交叉,内容系统完善。学校可根据教学计划从中选择教材的各种组合,使其适合本校的教学特点。

3. 掌握基础知识,侧重培养应用能力

目前社会对人才的需要更侧重于其应用能力。培养应用能力,须具备计算机基础理论、良好的综合素质和实践能力。理论知识作为基础必须掌握,本套教材通过实践教学与实例教学培养解决实际问题的能力和知识综合运用的能力。

4. 教学经验丰富的作者队伍

高等学校在计算机教学和教材改革上已经做了大量的工作,很多教师在计算机教育与科研方面积累了相当多的宝贵经验。本套教材均由有丰富教学经验的教师编写,并将这些宝贵经验渗透到教材中,使教材独具特色。

5. 配套资源完善

所有教材均配有 PPT 电子教案,部分教材配有实践教程、题库、教师手册、学习指南、习题解答、程序源代码、演示软件、素材、图书出版后要更新的内容等,以方便教与学。

我社致力于优秀教材的出版,恳请大家在使用的过程中,将发现的问题与提出的意见反馈给我们,以便再版时修改。

人民邮电出版社

自学读物。

本书由陈志泊编著。参加本书文字校对和多媒体课件制作的还有钱希志、翟巍巍、赵禹晶、鞠晓娜等，在此一并表示感谢。

书中不足之处，敬请批评指正。

编者

2008年10月

目 录

第 1 章 Visual Basic.NET 程序开发环境 1	
1.1 Visual Basic.NET 语言简介..... 1	
1.1.1 Visual Basic.NET 语言发展简介..... 1	
1.1.2 Visual Basic.NET 的版本..... 2	
1.1.3 Visual Basic.NET 语言的特点..... 2	
1.2 Microsoft.NET 框架概述..... 3	
1.3 Visual Studio 2005 集成开发环境的启动和退出..... 4	
1.4 Visual Studio 2005 集成开发环境的组成..... 7	
1.4.1 主窗口..... 7	
1.4.2 解决方案资源管理器窗口..... 9	
1.4.3 窗体设计窗口..... 12	
1.4.4 工具箱窗口..... 13	
1.4.5 属性窗口..... 14	
1.4.6 代码编辑窗口..... 15	
1.5 Visual Basic.NET 程序设计中的几个基本概念..... 17	
1.5.1 对象和类..... 17	
1.5.2 对象的组成要素..... 18	
1.5.3 控件..... 20	
1.6 Visual Basic.NET 程序设计的步骤和方法..... 20	
1.6.1 创建用户界面..... 21	
1.6.2 设置对象的属性值..... 24	
1.6.3 编写事件过程代码..... 27	
1.6.4 保存项目..... 28	
1.6.5 运行程序..... 28	
1.6.6 部署和安装应用程序..... 29	
习题..... 33	
第 2 章 Visual Basic.NET 语言基础 36	
2.1 Visual Basic.NET 的代码书写规则..... 36	
2.2 Visual Basic.NET 中的几种简单语句..... 36	
2.3 Visual Basic.NET 的数据类型..... 38	
2.4 命名空间..... 40	
2.5 常量与变量..... 41	
2.5.1 常量与符号常量..... 41	
2.5.2 变量及其命名规则..... 44	
2.5.3 变量的声明与初始化..... 44	
2.5.4 结构数据类型..... 47	
2.5.5 枚举数据类型..... 50	
2.6 Visual Basic.NET 的运算符和表达式..... 52	
2.6.1 算术运算符与算术表达式..... 52	
2.6.2 字符串连接运算符与字符串表达式..... 55	
2.6.3 关系运算符与关系表达式..... 55	
2.6.4 逻辑运算符和逻辑表达式..... 57	
2.6.5 赋值运算符与复合赋值运算符..... 59	
2.6.6 各种运算符的优先级比较..... 59	
2.7 Visual Basic.NET 的内部函数..... 60	
2.7.1 数学函数..... 61	
2.7.2 随机数函数..... 62	
2.7.3 转换函数..... 62	
2.7.4 日期和时间函数..... 63	
2.7.5 字符串处理函数..... 64	
2.7.6 数据类型转换函数..... 66	
习题..... 67	
第 3 章 Visual Basic.NET 可视化程序设计初步 70	
3.1 窗体的结构、常用属性、事件和方法..... 70	
3.1.1 窗体的结构..... 70	
3.1.2 窗体的常用属性..... 71	
3.1.3 窗体的常用事件..... 76	
3.1.4 窗体的常用方法..... 77	
3.2 命令按钮控件 Button 的常用成员及其使用..... 80	

3.3 标签控件 Label 的常用成员及其使用	83	第 6 章 过程	148
3.4 文本框控件 TextBox 的常用成员及其使用	85	6.1 过程的基本概念与分类	148
3.5 对象的输入焦点与 Tab 键序	90	6.2 Sub 过程的定义和调用	149
3.6 事件驱动应用程序	91	6.2.1 Sub 过程的定义	149
3.6.1 事件驱动的编程机制与程序代码的编写	91	6.2.2 Sub 过程的调用	150
3.6.2 事件驱动应用程序的工作方式	92	6.2.3 Sub 过程的建立方法	152
习题	92	6.3 Function 过程的定义和调用	154
第 4 章 Visual Basic.NET 程序控制结构	96	6.3.1 Function 过程的定义	154
4.1 顺序结构	96	6.3.2 Function 过程的调用	154
4.1.1 顺序程序结构	96	6.4 过程中的参数及其传递	156
4.1.2 InputBox 函数	97	6.4.1 形式参数与实际参数	156
4.1.3 MsgBox 函数	100	6.4.2 实际参数和形式参数的对应关系	157
4.2 选择结构	102	6.4.3 按值传递与按地址传递	158
4.2.1 单行结构 If 语句	102	6.4.4 数组参数的传递	160
4.2.2 块结构 If 语句	105	6.4.5 可选参数与可变参数	161
4.2.3 IIf 语句	107	6.5 变量的作用范围	164
4.2.4 Select Case 语句	108	习题	166
4.2.5 Switch 函数	110	第 7 章 常用基础控件	171
4.2.6 Choose 函数	111	7.1 提示控件 ToolTip	172
4.3 循环结构	112	7.2 单选按钮和复选框控件	174
4.3.1 While 循环结构	112	7.2.1 单选按钮控件 RadioButton	174
4.3.2 For 循环控制结构	114	7.2.2 复选框控件 CheckBox	174
4.3.3 Do 循环控制结构	119	7.3 容器工具控件	177
习题	124	7.3.1 框架控件 GroupBox	177
第 5 章 数组	130	7.3.2 面板控件 Panel	178
5.1 数组的概念	130	7.4 列表类控件	180
5.2 数组的声明及初始化	131	7.4.1 列表框控件 ListBox	180
5.2.1 数组的声明	131	7.4.2 复选列表框控件 CheckedListBox	185
5.2.2 数组的初始化	132	7.4.3 组合框控件 ComboBox	188
5.2.3 重定义数组大小	133	7.5 旋转控件 NumericUpDown 与 DomainUpDown 控件	190
5.2.4 结构类型的数组	134	7.5.1 NumericUpDown 控件	190
5.3 数组的引用与输入/输出	134	7.5.2 DomainUpDown 控件	191
5.4 For Each...Next 语句	138	7.6 滚动条控件	193
5.5 数组的综合应用举例	139	7.7 TrackBar 控件	196
习题	144	7.8 计时器控件 Timer	197
		7.9 图片框控件 PictureBox	199

习题	201	10.2.4 方法的定义及调用	247
第 8 章 菜单与工具栏设计	204	10.2.5 属性的定义与使用	250
8.1 菜单概述	204	10.2.6 事件的定义和使用	254
8.1.1 菜单的分类与结构	204	10.2.7 构造函数和析构函数	258
8.1.2 菜单项对象的常用属性和事件	205	10.3 类的继承与派生	259
8.2 MenuStrip 控件与下拉式菜单的创建	205	10.3.1 基类和派生类	259
8.2.1 MenuStrip 控件	205	10.3.2 派生类的构造函数	260
8.2.2 利用 MenuStrip 控件创建下拉式 菜单	207	10.4 类的多态性	262
8.3 ContextMenuStrip 控件与弹出式菜单 的创建	210	10.4.1 重载与重写	262
8.4 工具栏与 ToolStrip 控件	212	10.4.2 多态性及其实现	265
8.4.1 ToolStrip 控件	212	习题	267
8.4.2 利用 ToolStrip 控件设计 工具栏	213	第 11 章 文件	269
习题	216	11.1 文件及其结构	269
第 9 章 对话框	218	11.1.1 文件的概念与命名	269
9.1 对话框的基本概念及分类	218	11.1.2 文件的结构	270
9.2 文件对话框	218	11.1.3 文件的分类	270
9.3 “颜色”对话框	221	11.2 文件的访问	271
9.4 “字体”对话框	222	11.2.1 文件的访问步骤	271
9.5 多重窗体	224	11.2.2 文件的访问方法	271
9.5.1 添加窗体与设置启动窗体	224	11.3 使用 System.IO 命名空间中的类 访问文件	272
9.5.2 窗体的实例化与显示	225	11.3.1 有关流的基本概念	272
9.5.3 不同窗体间数据的访问	227	11.3.2 使用 FileStream 类访问文件	273
9.5.4 多重窗体程序设计的方法和步骤	228	11.3.3 使用 StreamReader 和 Stream Writer 类访问文本文件	276
9.6 键盘与鼠标事件	232	11.3.4 使用 BinaryReader 和 Binary Writer 类访问二进制文件	279
9.6.1 键盘事件	232	习题	283
9.6.2 鼠标事件	235	第 12 章 Visual Basic.NET 图形 程序设计	285
习题	237	12.1 GDI+绘图概述	285
第 10 章 Visual Basic.NET 面向 对象程序设计	241	12.1.1 GDI+的有关概念	285
10.1 面向对象程序设计的基本特性	241	12.1.2 坐标系	286
10.2 类和对象的定义	243	12.1.3 利用 GDI+绘制图形的方法 步骤	286
10.2.1 类定义的语法格式	243	12.2 与绘图相关的对象及使用	286
10.2.2 在类中定义数据成员	245	12.2.1 System.Drawing 命名空间中的 常用类和结构	286
10.2.3 对象的定义及其成员的访问	246		

12.2.2 Graphics 对象.....	287	13.2.5 ADO.NET 的联机与脱机数据 存取模式.....	307
12.2.3 画笔 Pen 对象及使用.....	291	13.2.6 ADO.NET 开发数据库应用程序 的一般步骤.....	307
12.2.4 画刷 Brush 对象及使用.....	294	13.3 ADO.NET 对象及其编程.....	308
12.2.5 字体 Font 对象及使用.....	297	13.3.1 创建和使用 Connection 对象.....	308
习题.....	299	13.3.2 创建和使用 Command 对象.....	310
第 13 章 Visual Basic.NET 数据库 管理应用程序开发.....	301	13.3.3 创建和使用 DataReader 对象.....	311
13.1 数据库的基本概念.....	301	13.3.4 创建和使用 DataSet 对象.....	313
13.1.1 关系数据库与二维表.....	301	13.3.5 创建和使用 DataTable 对象.....	315
13.1.2 关系数据库的有关概念.....	302	13.3.6 创建和使用 DataAdapter 对象.....	317
13.1.3 关系数据库的操作.....	302	13.3.7 数据绑定与 DataGridView 控件.....	321
13.2 ADO.NET 简介.....	304	13.3.8 数据库的编辑.....	325
13.2.1 ADO.NET 体系结构.....	304	习题.....	330
13.2.2 .NET 数据提供者.....	304	参考文献.....	332
13.2.3 DataSet 对象.....	306		
13.2.4 ADO.NET 相关类的命名空间.....	306		

第 1 章

Visual Basic.NET 程序开发环境

1.1 Visual Basic.NET 语言简介

1.1.1 Visual Basic.NET 语言发展简介

BASIC 是英文 Beginner's All-purpose Symbolic Instruction Code(初学者通用的符号指令代码)的缩写,它是由美国的达特茅斯学院(Dartmouth College)数学系的 John G. Kemeny 和 Thomas E. Kurtz 两位教授于 1963~1964 年间开发的一种小型程序设计语言。当时它的主要用途是教学,后来因其简单易学,接近美国口语,程序运行调试容易,很快得到了广泛的应用。20 世纪 80 年代,随着结构化程序设计的需要,新版本的 BASIC 语言在功能上有了较大扩充,增加了数据类型和程序控制结构,其中较有影响的有 True Basic、Quick Basic 和 Turbo Basic。

1988 年,Microsoft 公司推出了用于个人计算机的基于图形用户界面(Graphics User Interface, GUI)的 Windows 操作系统,特别是 1990 年 5 月 Windows 3.0 操作系统推出后,在全世界范围内引起了个人计算机突飞猛进的发展。在基于图形用户界面的 Windows 操作系统下,用户只要通过鼠标的单击和拖动即可形象地完成各种操作,操作简单方便,不必人工记忆和输入复杂的操作系统命令。因此,相对于原来的基于字符用户界面的 DOS 操作系统来说,基于图形用户界面的 Windows 操作系统更受用户和软件开发商的欢迎。

随着操作系统的改变,程序运行机理和程序设计方式也发生了改变。对于普通计算机用户来说,要开发出基于 Windows 操作系统的应用程序是相当困难的,因为程序员要完成的事情非常多,要将很多精力放在开发 GUI(图形用户界面)上,如控制鼠标的位置、确定用户是单击还是双击鼠标等,这使很多希望学习 Windows 开发的编程人员望而却步。因此,如何找一种简单、易学、好用的程序设计语言开发在 Windows 操作系统下的应用程序变得越来越重要、越来越迫切。

于是,Microsoft 公司从 1991 年起就推出了 Visual Basic 1.0 版来简化 Windows 应用程序的开发。Visual Basic 的意思是“可视化的 BASIC”,是对原来 BASIC 语言的扩充。它既保留了 BASIC 语言简单易用的优点,又充分利用了 Windows 提供的图形界面,相当于是一个崭新的可视化设计工具。任何一个有初步程序设计基础的人,都能够在很短的时间内掌握 Visual Basic,并用它编写出在 Windows 下运行的各种应用程序。

可以说,Windows 操作系统的产生呼唤着 Visual Basic 的问世,同时,Visual Basic 的产生大大推动了 Windows 应用程序的开发工作。Visual Basic 语言已经不是以前意义上的 BASIC 语言了,

它的功能已经变得十分强大，可以胜任任何大型应用程序的开发工作。

1.1.2 Visual Basic.NET 的版本

从时间（即纵向）上看，微软公司开发的 Visual Basic 大致经历了以下几个发展过程。

- 1991 年推出了 Visual Basic 1.0 版本。
- 1992 年经过对 Visual Basic 1.0 版本的修改后，推出了 Visual Basic 2.0 版本。
- 1993 年再次修改完善后，推出了 Visual Basic 3.0 版本。
- 1995 年推出了 Visual Basic 4.0 版本。
- 1997 年发布了 Visual Basic 5.0 版本。
- 1998 年推出了 Visual Basic 6.0 版本。
- 2002 年发布了 Visual Basic.NET 2002 版本。
- 2003 年发布了 Visual Basic.NET 2003 版本。
- 2005 年发布了 Visual Basic.NET 2005 版本。

本书将以 Visual Basic.NET 2005 中文版为例进行讲解。

Visual Basic.NET 2005 不是一个独立的开发工具，而是与 Visual C++、Visual C#、Visual J#（简称 C++、C#、J#）等一起集成在 Visual Studio 2005 中，是 Visual Studio 2005 的编程套件之一。

依靠 Microsoft .NET Framework（简称.NET 框架）的支持，Visual Basic.NET 成为开发 Windows 应用程序和基于 Web 的 ASP.NET 应用程序的主要工具之一。

1.1.3 Visual Basic.NET 语言的特点

由于 Visual Basic.NET 是在原来 BASIC 语言的基础上开发而成的，因此它具有原 BASIC 语言简单、易学易用的优点，同时它又增加了结构化和可视化程序设计语言的功能。总的来说，Visual Basic.NET 是一种可视化的、面向对象和采用事件驱动编程机制的结构化高级程序设计语言，它有以下几个主要特点。

（1）编程的可视化。程序开发人员可利用 Visual Studio 2005 提供的可视化设计工具，在窗体中画出各种“组件”图形对象，并利用其提供的工具设置这些图形对象的属性，即可设计好程序的用户界面。用户不需要去编写大量的程序代码实现用户界面，大大提高了程序设计的效率，同时整个过程是“所见即所得”的，即程序设计人员在设计过程中就能清楚地看到所设计的程序界面是否满意，设计过程非常直观、形象。

（2）面向对象的程序设计方法。Visual Basic.NET 全面支持面向对象的程序设计方法（OOP），即满足封装性、继承性和多态性。这种程序设计方法把程序代码和数据封装起来作为一个对象，以对象作为计算主体，所有的编程操作都是围绕着对象展开的。

（3）事件驱动的编程机制。程序代码的执行是由有关某一对象的事件的发生来驱动的，Visual Basic.NET 给每个对象规定了它所能响应的的事件。在程序执行的过程中，当某一对象的某事件发生时，有关该事件的程序代码将被执行。所以，用户在编写程序时，只需要编写最关心的事件发生时所需要执行的程序代码段即可。

（4）支持结构化程序设计。Visual Basic.NET 提供的控制结构完全支持传统的结构化程序设计，可以编制结构清晰简明的程序。

（5）强大的数据库功能。Visual Basic.NET 采用 ADO.NET 数据访问技术。对各种不同类型的数据库，如 Access、Microsoft SQL Server、Oracle 等数据库，都以统一的方式管理和访问数据源

中的数据，使用户能在很短的时间内编写出数据库管理系统。

(6) 具有丰富的数据类型、功能强大的类库。Visual Basic.NET 不仅拥有与 C++ 同样丰富的数据类型，而且由于将数据类型定义成类，因而数据类型本身也提供了数据处理的能力；依靠 .NET 框架的支持，Visual Basic.NET 程序几乎可以获取 Windows 提供的所有功能。

(7) 具有强大的网络功能。Visual Basic.NET 中有 Web Service，它是 DCOM 的取代者。另外，Visual Basic.NET 中还有 Web Forms，它可以使用户无须使用 ASP 或 CGI 就能有效地建立全交互的互联网网站。

1.2 Microsoft.NET 框架概述

随着 Internet 的不断发展和广泛应用，Internet 逐渐成为编程领域的中心，于是 Microsoft 公司在 2002 年发布了在技术上具有革命性意义的网络计算平台——Microsoft.NET（即 .NET 战略），其目的是将不同平台与不同系统加以集成，这种集成服务平台允许各种系统环境下的应用程序通过互联网进行通信和共享数据。用 Microsoft 公司 CEO 史蒂夫·巴尔默的话来说，.NET 是“代表了一个集合、一个环境、一个编程的基本结构，作为一个平台来支持下一代的 Internet。 .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、服务器或任何地方，与该编程模式具有很好的一致性，并有新的创意。因此，它不仅是一个用户体验，而且是开发人员体验的集合，这就是对 .NET 的概念性描述”。

.NET 框架和 .NET 开发工具一起构成了完整的 .NET 开发平台。

.NET 开发工具包括 Visual Studio 2005 集成开发环境和 .NET 开发语言，其中，Visual Studio 2005 集成开发环境用于开发和测试应用程序，.NET 开发语言包括 Visual Basic 2005、Visual C# 2005、Visual C++ 2005、Visual J# 2005。

.NET 框架具有两个主要组件：公共语言运行库（Common Language Runtime, CLR）和 .NET Framework 基础类库。其结构及其与 Visual Studio 开发工具之间的关系如图 1.1 所示。

现将 .NET 框架的主要组成部分说明如下。

1. CLR

CLR 是 .NET 框架的运行环境，位于 .NET 框架的底层，为基于 .NET 平台的一切操作提供一个统一的、受控的运行环境，提供程序代码的编译、安全性的控制和管理、链接库与执行文件的缓存等功能。

面向 .NET 框架的代码称为托管代码（Managed Code）。托管代码在执行过程中完全被 CLR 所控制，CLR 负责执行程序，提供内存管理、线程管理、安全管理、代码验证和编译、异常处理、通用类型系统与生命周期监控等核心服务。凡是符合公共语言规范的程序语言（如 C# 和 Visual Basic.NET 等）所开发的程序都可以在任何有 CLR 的操作系统上执行。在某种程度上，CLR 和 Java 虚拟机的原理基本一致。

.NET 框架利用 CLR 解决了各种语言的 Runtime（执行期）不可共享问题，具有跨平台特性。

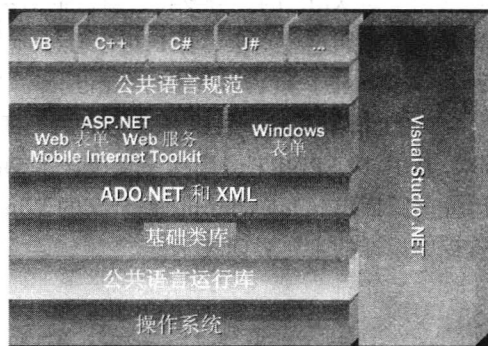


图 1.1 .NET 框架结构及其与 Visual Studio 间的关系

Runtime 是指计算机编译应用程序的运行时（状态），它包括编译语言所需的函数和对象等，因此，不同编程语言的 Runtime 是不同的，各种语言间的 Runtime 不能共享。.NET 框架以 CLR 解决了这个共享问题，它以中间语言（Intermediate Language, IL）实现程序转换。IL 是介于高级语言和机器语言间的中间语言，包括对象加载、方法调用、流程控制、逻辑运算等多种基本指令。在 .NET 框架之上，无论是采用哪种编程语言编写的程序，都被编译成 IL，IL 经过再次编译形成机器码，完成 IL 到机器码编译任务的是 JIT（Just In Time）。这一处理过程如图 1.2 所示。

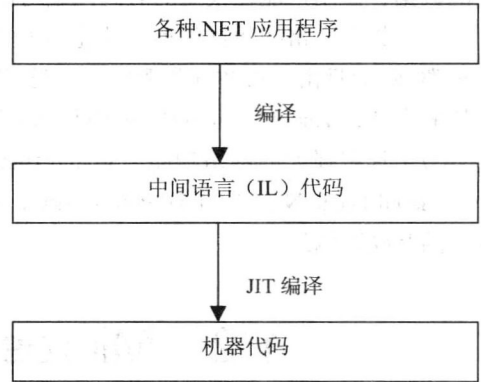


图 1.2 .NET 应用程序的编译过程

2. .NET Framework 基础类库

在 .NET 框架结构中，.NET Framework 类库（.NET Framework Class Library）位于 CLR 之上，包含许多高度可重用性的接口和类，并且完全面向对象。它既是 .NET 应用软件开发的基础类库，也是 .NET 平台本身的实现基础。该类库以命名空间（Namespace）方式来组织，最顶层的命名空间是 System。命名空间与类库的关系就像文件系统中的目录与文件的关系一样。关于命名空间的概念与使用将在后续章节中详细介绍。

3. ADO.NET

ADO.NET 为 .NET 框架提供统一的数据访问技术。与以前的数据访问技术相比，ADO.NET 主要增加了对 XML 的充分支持、新数据对象的引入、语言无关对象的引入以及使用和 CLR 一致的类型等，利用这些对象，用户可以轻松地完成对数据库的复杂操作。

4. ASP.NET 和 Windows Forms

在 .NET 框架基础上可以开发的应用程序主要包括 ASP.NET 应用程序和 Windows Forms 应用程序，其中，ASP.NET 应用程序又包含了 Web Forms 和 Web Service，它们组成了全新的因特网应用程序；而 Windows Forms 是全新的 Windows 窗口式应用程序。可见，它们是在 .NET 框架下进行程序设计的主要界面技术。

1.3 Visual Studio 2005 集成开发环境的启动和退出

1. Visual Studio 2005 的启动

要使用 Visual Basic.NET 开发程序，必须先进入 Visual Studio 2005 集成开发环境，方法如下。

在 Windows 系统下单击“开始”按钮，执行“程序”|“Microsoft Visual Studio 2005”|“Microsoft Visual Studio 2005”命令，即可打开 Microsoft Visual Studio 2005 集成开发环境界面，如图 1.3 所示。

2. 进入 Visual Studio 2005 集成开发环境

在图 1.3 所示的画面中，“最近的项目”显示的是用户最近创建的项目，可以直接单击其中某个项目，系统会自动打开该项目并进入 Visual Studio 2005 集成开发环境（IDE）。



图 1.3 Microsoft Visual Studio 2005 集成开发环境

若需要新建一个项目并进入 Visual Studio 2005 集成开发环境，则可按以下步骤进行。

(1) 在图 1.3 所示的界面中，执行“文件”|“新建项目”菜单命令，出现如图 1.4 所示的“新建项目”对话框。

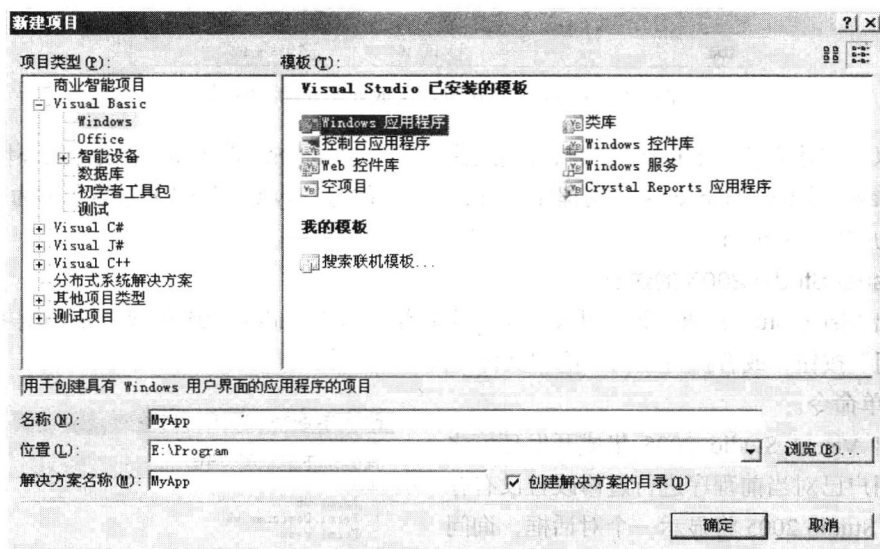


图 1.4 “新建项目”对话框

(2) 在图 1.4 所示的对话框左侧“项目类型”列表框中单击并选择“Visual Basic”（由于要创建的项目是用 Visual Basic 语言开发的），在对话框右侧的“模板”列表框显示了可以创建的应用程序类型，以满足用户开发不同程序的需要，如 Windows 应用程序、控制台应用程序、类库等。由于用户要创建的程序是可以在 Windows 下运行的窗口式的应用程序，所以从中选择“Windows 应用程序”。

在对话框的下部“名称”文本框中显示了系统给当前项目的默认名称“WindowsApplication1”，用户可以使用该默认名称，也可重新命名，此处将项目名称改为“MyApp”。

在“位置”下拉列表框中显示了项目保存的位置，默认为 Visual Studio 2005 系统所在目录，此处将项目存放的位置改为“E:\Program”（用户视自己的磁盘目录结构而定），或单击右侧的“浏览”按钮选择项目存放的位置。

在“解决方案名称”文本框中显示了系统给定的解决方案的默认名称“WindowsApplication1”，用户可以使用该默认名称，也可重新命名，此处将解决方案名称改为“MyApp”。

(3) 单击“确定”按钮，此时就新创建了一个 Windows 应用程序项目，并向该项目中自动添加一个名称为 Form1.vb 的窗体文件，然后进入到 Visual Studio 2005 集成开发环境，如图 1.5 所示。

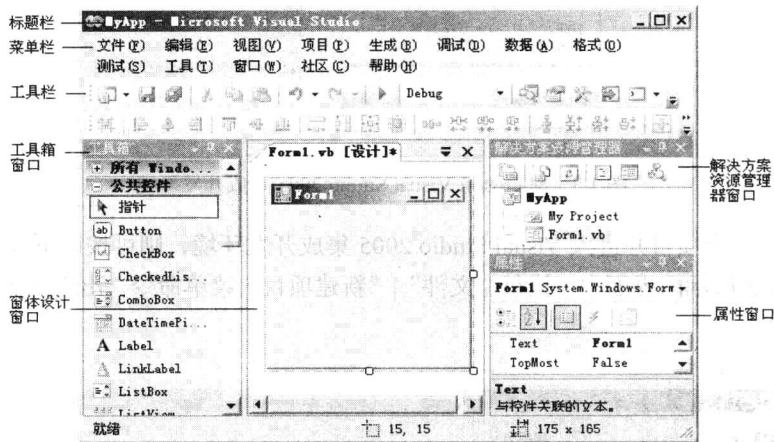


图 1.5 Visual Studio 2005 集成开发环境界面图

为了以后叙述方便，上述操作步骤可描述为：进入 Visual Studio 2005 集成开发环境，新建一个 Visual Basic 类型的 Windows 应用程序项目，项目名称为“MyApp”，解决方案名称为“MyApp”，存放位置为“E:\Program”。

3. Visual Studio 2005 的退出

要退出 Visual Studio 2005 集成开发环境，可单击 Visual Studio 2005 集成开发环境主窗口右上角的“关闭”按钮，或者执行主窗口的“文件”|“退出”菜单命令。

在退出 Visual Studio 2005 集成开发环境之前，如果用户已对当前程序进行过修改且没有存盘，Visual Studio 2005 将显示一个对话框，询问用户是否将其存盘，如图 1.6 所示。

如果单击“是”按钮，则对当前程序进行保存，如果单击“否”按钮，则系统会自动放弃对当前程序的保存，两种操作情况下都可退出 Visual Studio 2005 集成开发环境。如果单击“取消”按钮，则系统不退出集成开发环境，而是返回到系统原有状态。

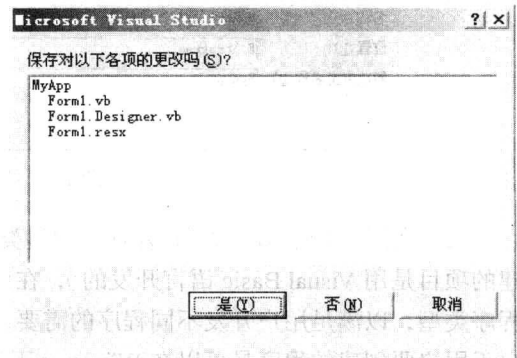


图 1.6 提示用户是否进行保存对话框