



软件职业技术学院“十一五”规划教材

Visual C# 2008

程序设计

主 编 曹 静 副主编 陈 娜 谢日星



中国水利水电出版社
www.waterpub.com.cn

规划教材

Visual C# 2008 程序设计

陈娜 谢日星



中国水利水电出版社

www.waterpub.com.cn


www.waterpub.com.cn

内 容 提 要

Visual C#已成为在.NET 平台上进行开发的首选语言。Visual C# 2008 融 C++的灵活性和强大功能与 Visual C#的简单性于一身。本书通过开发一个客户信息管理程序,全面阐述了使用 Visual C# 2008 开发应用程序的基本知识。

本书分为 10 章。主要内容包括 Visual C# 2008 速成版开发环境;变量和数据类型以及表达式的使用;三种典型的程序结构;数组的使用;类和对象的使用;类和对象的高级应用及 C# 3.0 的新特性;集合和泛型;异常处理和程序调试;文件和流;基于 Windows 的程序设计。

本书通过简洁的语言和详细的步骤,帮助读者迅速掌握使用 Visual C# 2008 开发应用程序所需要的基本知识。本书适合没有任何编程经验的读者和 Visual C#新手阅读,也可作为本专科院校的学生学习 Visual C#编程参考用书。通过对本书的学习,读者可循序渐进地掌握 C#编程技术,从而开发出优秀的应用程序。

本书配有电子教案,读者可以从中国水利水电出版社网站和万水书苑免费下载,网址为:
<http://www.waterpub.com.cn/softdown/>和 <http://www.wsbookshow.com>。

图书在版编目 (C I P) 数据

Visual C# 2008程序设计 / 曹静主编. — 北京 :
中国水利水电出版社, 2010.7
软件职业技术学院“十一五”规划教材
ISBN 978-7-5084-7636-0

I. ①V… II. ①曹… III. ①
C语言—程序设计—高等学校:技术学校—教材 IV
①TP312

中国版本图书馆CIP数据核字(2010)第120788号

策划编辑:杨庆川 责任编辑:杨元泓 加工编辑:周益丹 封面设计:李 佳

书 名	软件职业技术学院“十一五”规划教材 Visual C# 2008 程序设计
作 者	主 编 曹 静 副主编 陈 娜 谢日星
出版发行	中国水利水电出版社 (北京市海淀区玉渊潭南路 1 号 D 座 100038) 网址: www.waterpub.com.cn E-mail: mchannel@263.net (万水) sales@waterpub.com.cn 电话: (010) 68367658 (营销中心)、82562819 (万水)
经 售	全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	北京蓝空印刷厂
规 格	184mm×260mm 16 开本 18.5 印张 452 千字
版 次	2010 年 7 月第 1 版 2010 年 7 月第 1 次印刷
印 数	0001—3000 册
定 价	32.00 元

凡购买我社图书,如有缺页、倒页、脱页的,本社营销中心负责调换

版权所有·侵权必究

序

随着信息技术的广泛应用和互联网的迅猛发展,以信息产业发展水平为主要特征的综合国力竞争日趋激烈,软件产业作为信息产业的核心和国民经济信息化的基础,越来越受到世界各国的高度重视。中国加入世贸组织后,必须以积极的姿态,在更大范围和更深程度上参与国际合作和竞争。在这种形势下,摆在我们面前的突出问题是人才短缺,计算机应用与软件技术专业领域技能型人才的缺乏尤为突出,无论是数量还是质量,都远不能适应国内软件产业的发展和信息化建设的需要。因此,深化教育教学改革,推动高等职业教育与培训的全面发展,大力提高教学质量,是迫在眉睫的重要任务。

2000年6月,国务院发布《鼓励软件产业和集成电路产业发展的若干政策》,明确提出鼓励资金、人才等资源投向软件产业,并要求教育部门根据市场需求进一步扩大软件人才培养规模,依托高等学校、科研院所,建立一批软件人才培养基地。2002年9月,国务院办公厅转发了国务院信息化工作办公室制定的《振兴软件产业行动纲要》,该《纲要》明确提出要改善软件人才结构,大规模培养软件初级编程人员,满足软件工业化生产的需要。教育部也于2001年12月在35所大学启动了示范性软件学院的建设工作,并于2003年11月启动了试办示范性软件职业技术学院的建设工作。

示范性软件职业技术学院的建设目标是:经过几年努力,建设一批能够培养大量具有竞争能力的实用型软件职业技术人才的基地,面向就业、产学结合,为我国专科层次软件职业技术人才培养起到示范作用,并以此推动高等职业技术教育人才培养体系与管理体制和运行机制的改革。要达到这个目标,建立一套适合软件职业技术学院人才培养模式的教材体系显得尤为重要。

高职高专的教材建设已经走过了几个发展阶段,由最开始本科教材的压缩到加大实践性教学环节的比重,再到强调实践性教学环节,但是学生在学习时还是反映存在理论与实践的结合问题。为此,中国水利水电出版社在经过深入调查研究后,组织了一批长期工作在高职高专教学一线的老师,编写了这套“软件职业技术学院‘十一五’规划教材”,本套教材采用项目驱动的方法来编写,即全书所有章节都以实例作引导来说明各知识点,各章实例之间并不是孤立的,每个实例都可以作为最终项目的一个组成部分;每一章章末还配有实习实训(或叫实验),这些实训组合起来是一个完整的项目。

采用这种方式编写的图书与市场上同类教材相比更具优越性,学生不仅仅学到了知识点,还通过项目将这些知识点连成一条线,开拓了思路,掌握了知识,达到了面向岗位的职业教育培训目标。

本套教材的主要特点有:

- (1) 课程主辅分明——重点突出,教学内容实用。
- (2) 内容衔接合理——完全按项目运作所需的知识体系结构设置。

(3) 突出实习实训——重在培养学生的专业能力和实践能力,力求缩短人才与企业间的磨合期。

(4)教材配套齐全——本套教材不仅包括教学用书,还包括实习实训材料、教学课件等,使用方便。

本套教材适用于广大计算机专业和非计算机专业的大中专院校的学生学习,也可作为有志于学习计算机技术与开发的工程技术人员参考教材。

编委会

2006 年 7 月

前 言

Visual C#作为微软的旗舰编程语言,经过几年的长足发展,在全球得以迅速普及,成为很多程序员的首选语言。Visual C# 2008 新增了大量可圈可点的丰富特性,本书围绕着基础知识和这些新特性,全面介绍了如何利用 Visual Studio 2008 和 .NET Framework 来进行 C#编程。

本书是全国示范性软件职业学院计算机及其相关专业指定教材,针对全国示范性软件职业学院特点,以够用为度,淡化理论,强化技能,重在实际操作,在完成必要的理论阐述之后,以实际的代码案例来解释理论知识,适合于熟悉计算机基础知识的学生作为教材或自学用书。

全书以实际项目设计贯穿全书,在每章结束后,根据本章所学知识点提出项目要求,在实训环节引导读者从无到有完成一个实训项目,最后形成一个客户管理系统,让学生能在实践中掌握 Visual C# 2008 程序设计。本书是一本有效、实用的入门级教材。适合于学习 Visual C# 2008 技术的初学者。

本书是作者在多年的教学实践、科学研究以及项目实践的基础上,参阅了大量国内外相关教材后,几经修改而成。主要特点如下:

1. 语言严谨、精练

对基本概念和技术进行了清楚准确的解释并结合实例说明,让读者能较轻松地掌握每一个知识点。

2. 实际项目开发与理论教学紧密结合

为了使读者能快速地掌握 Visual C# 2008 的相关技能并熟练运用,本书根据每章节的知识点设计了一个实训项目——客户信息管理,每章节为实训增加或改进一部分功能,学完本书后,实训是一个完整的项目。

3. 合理、有效的组织

本书按照由浅入深的顺序,循序渐进、系统地介绍程序设计的相关知识和技能。各个章节的编写以实践应用为目标,理论的阐述主要围绕着实际应用技术组织和展开,练习的重要性得到体现,不再附属于相关理论知识。

4. 内容充实、实用

本书的练习紧紧围绕着实际项目进行,在各章完成各种技术准备和练习后,为完成项目掌握了基本的知识点,只需按照书本中的实训步骤,利用所学习的知识点,即可完成项目要求。

5. 本书配有全部的程序源文件和电子教案

为方便读者使用,书中全部实例的源代码及电子教案均免费赠送给读者,读者可以从中国水利水电出版社网站和万水书苑免费下载,网址为: <http://www.waterpub.com.cn/softdown/>和 <http://www.wsbookshow.com>。

本书以最新的 .NET 3.5 和 Visual C# 2008 为基础循序渐进地介绍 Visual C# 2008 入门所需的各方面知识,包括开发环境的配置、C#语法、Windows 应用程序开发、处理文件以及 .NET

3.5 的新增特性等。

本书同时还介绍了大量 Visual C# 2008 的开发经验,对使用中的重点、难点进行了专门的讲解,本书由曹静担任主编,陈娜、谢日星担任副主编,董宁、陈丹、罗炜、杨威、张松慧、肖英、江俊、赵丙秀、刘洁、李文慧、刘嵩、肖奎、余恒芳、郭俐、库波参加了本书的编写工作,陈娜、谢日星统编全稿。

由于时间仓促,加之编者水平有限,书中不妥或错误之处在所难免,殷切希望广大读者批评指正。同时,恳请读者一旦发现错误,于百忙之中及时与编者联系,以便尽快更正,编者将不胜感激。E-mail: Jingcao_2002@163.com

编者

2010年6月

序	1
前言	1
第1章 C#与 Visual Studio 2008	1
本章学习目标	1
1.1 .NET 简介	1
1.2 Visual Studio 2008 简介	3
1.3 Visual C#.NET 集成开发环境简介	6
1.4 创建第一个 C#控制台（或称命令行）程序	19
1.5 创建第一个 C# Windows 程序	21
1.6 C#程序结构介绍	22
本章小结	30
习题	31
案例完善——客户信息管理系统中的新建项目	31
第2章 变量、数据类型和表达式	33
本章学习目标	33
2.1 注释	33
2.2 标识符	33
2.3 常量和变量	34
2.4 数据类型	35
2.4.1 值类型	36
2.4.2 引用类型	41
2.4.3 隐含类型	41
2.5 类型转换	42
2.5.1 自动类型转换	42
2.5.2 强制类型转换	43
2.5.3 字符串与基本数据类型间的转换（基本数据类型间的转换）	43
2.6 运算符	44
2.6.1 运算符的分类	45
2.6.2 运算符的优先级	45
2.6.3 算术运算符	46
2.6.4 赋值运算符	47

录

2.6.5 关系运算符 48

2.6.6 位运算符 49

2.6.7 逻辑运算符 51

2.6.8 条件（三目）运算符 53

本章小结 53

习题 54

案例完善——客户信息管理系统中的客户

 信息描述 55

第3章 结构化程序设计语句 59

 本章学习目标 59

 3.1 程序的三种结构 59

 3.1.1 顺序结构 59

 3.1.2 分支结构 59

 3.1.3 循环结构 60

 3.2 if 语句 61

 3.3 switch 语句 68

 3.4 while 语句 70

 3.5 do-while 语句 72

 3.6 for 语句 74

 3.7 break 和 continue 语句 79

 本章小结 81

 习题 81

 案例完善——客户信息管理系统中的程序

 流程控制 83

第4章 数组 92

 本章学习目标 92

 4.1 数组的概念 92

 4.2 声明和创建数组 92

 4.3 初始化数组变量 94

 4.4 遍历数组元素 95

 4.5 多维数组 98

4.6 交错数组	100	案例完善——客户信息管理系统中的抽象类	
4.7 隐式类型数组	102	与类的继承	195
本章小结	103	第7章 集合和泛型	198
习题	103	本章学习目标	198
案例完善——客户信息管理系统中的多客户		7.1 集合与集合接口	198
信息管理	104	7.1.1 ArrayList 集合	198
第5章 面向对象	106	7.1.2 哈希表 Hashtable	200
本章学习目标	106	7.1.3 集合接口	202
5.1 面向对象程序设计概述	106	7.2 泛型集合	203
5.2 类的定义和对象的创建	107	7.2.1 泛型 List 集合	203
5.3 类的字段和属性	109	7.2.2 泛型 Stack 集合	204
5.4 索引器	116	7.2.3 泛型 Queue 集合	205
5.5 方法定义及调用	119	本章小结	206
5.6 值类型与引用类型	124	习题	206
5.7 参数的传递	127	案例完善——客户信息管理系统中的泛型	206
5.8 变量的作用域	132	第8章 程序的生成、异常处理和调试	209
5.9 构造函数	135	本章学习目标	209
5.10 静态成员	140	8.1 异常处理	209
5.11 内部类和匿名类	146	8.1.1 异常类	209
本章小结	147	8.1.2 try-catch	210
习题	147	8.1.3 try-finally	213
案例完善——客户信息管理系统中面向对象		8.1.4 try-catch-finally	214
的客户信息管理	148	8.1.5 多重 try 结构	216
第6章 面向对象的高级应用及 C#语法新特性	150	8.1.6 默认异常处理	217
本章学习目标	150	8.1.7 用户自定义异常	217
6.1 类的继承	150	8.1.8 throw	219
6.2 访问控制符	155	8.2 Visual Studio 2008 的调试功能	220
6.3 多态性	162	8.2.1 单步调试程序	220
6.4 密封类	168	8.2.2 Debugger 与 Debug 类的使用	222
6.5 抽象类	169	8.2.3 使用 Trace 类	225
6.6 接口	170	本章小结	227
6.7 委托与事件	173	习题	228
6.8 匿名类型	183	案例完善——客户信息管理系统中的异常	
6.9 对象与集合初始化器	185	处理	228
6.10 扩展方法	187	第9章 流与文件	230
6.11 Lambda 表达式	190	本章学习目标	230
本章小结	194	9.1 流的基本概念	230
习题	194	9.2 目录	231

9.2.1 DriveInfo 类	231	10.1 Windows 窗体应用程序概述	247
9.2.2 Directory 类	233	10.2 Windows 窗体及控件介绍	250
9.2.3 DirectoryInfo 类	235	10.3 常用控件的属性、方法和事件	253
9.3 File 类和 FileInfo 类	237	10.3.1 控件共有的属性、事件和方法	254
9.4 文件的读写	241	10.3.2 常用控件介绍	256
本章小结	245	10.3.3 常用控件的典型用法	258
习题	245	10.4 基于 Windows Forms 的程序设计	267
案例完善——客户信息管理系统中的数据 保存	245	本章小结	284
第 10 章 基于 Windows 的应用程序	247	习题	284
本章学习目标	247	案例完善——客户信息管理系统	284
		参考文献	286

第1章



C#与 Visual Studio 2008

本章学习目标

本章主要介绍 .NET 基础知识, Visual Studio 2008 集成开发环境的使用, 如何创建一个控制台应用程序, 以及如何创建一个 Windows 应用程序。通过对本章的学习, 读者应该掌握以下内容:

- Visual Studio 2008 集成开发环境的使用
- 创建控制台应用程序
- 创建简单的 Windows 应用程序
- 控制台应用程序的结构

1.1 .NET 简介

.NET 就是 Microsoft 的 XML Web 服务平台。Microsoft .NET 平台包含广泛的产品系列, 它们都是基于 XML 和 Internet 行业标准构建, 不论操作系统或编程语言有何差别, XML Web 服务能使应用程序在 Internet 上传输和共享数据。

.NET Framework 是构成 Microsoft .NET 平台核心部分的一组技术, 它为开发 Web 应用程序和 XML Web Services 提供了基本的构建模块。NET Framework 为创建和运行 .NET 应用程序提供了必要的编译和运行基础。

.NET 框架由 Windows Server System 构建, 部署与运行 Web 服务与应用程序的编程模型。该框架托管了大部分深层结构, 让开发人员专注于撰写应用程序的业务逻辑代码。

.NET Framework 是支持生成和运行下一代应用程序和 XML Web Services 的内部 Windows 组件。NET Framework 旨在实现下列目标:

- 提供一个一致的面向对象的编程环境, 而无论对象代码是在本地存储和执行, 还是在本地执行在 Internet 上分布, 或者是在远程执行的。
- 提供一个将软件部署和版本控制冲突最小化的代码执行环境。
- 提供一个可提高代码(包括由未知的或不完全受信任的第三方创建的代码)执行安全性的代码执行环境。
- 提供一个可消除脚本环境或解释环境的性能问题的代码执行环境。
- 使开发人员的经验在面对类型大不相同的应用程序(如基于 Windows 的应用程序和基于 Web 的应用程序)时保持一致。

- 按照工业标准生成所有通信, 以确保基于 .NET Framework 的代码可与任何其他代码集成。

.NET Framework 具有两个主要组件: 公共语言运行库和 .NET Framework 类库。公共语言运行库是 .NET Framework 的基础。可以将运行库看作一个在执行时管理代码的代理, 它提供内存管理、线程管理和远程处理等核心服务, 并且还强制实施严格的类型安全以及可提高安全性和可靠性的其他形式的代码准确性。事实上, 代码管理的概念是运行库的基本原则。以运行库为目标的代码称为托管代码, 而不以运行库为目标的代码称为非托管代码。 .NET Framework 的另一个主要组件是类库, 它是一个综合性的面向对象的可用类型集合, 可以使用它开发多种应用程序, 这些应用程序包括传统的命令行或图形用户界面 (GUI) 应用程序, 也包括基于 ASP.NET 所提供的最新创新的应用程序 (如 Web 窗体和 XML Web Services)。

1. 公共语言运行库

CLI (Common Language Infrastructure, 通用语言框架) 实际上是一种为虚拟机环境而制定的规范, 使得由各种高级语言所编制的程序可以在不同的系统环境中执行而不必更改或重新编译原程序代码。

微软的 .NET 的基础 CLR 就是 CLI 的一个实例。CLR 只不过是 CLI 规范在个人计算机、在微软 Windows 操作系统中的一个执行而已。毫无疑问, 其他操作系统环境和硬件平台上, CLI 也同样可行。你有时会发现术语 CLI 和 CLR 可交换使用, 尽管很明显它们不是一回事。CLI 是一种标准规范, 而 CLR 却是微软对 CLI 的实现。

公共语言运行库 (Common Language Runtime, CLR) 也叫公共语言运行时, 是 .NET 框架的基础。公共语言运行库简化了应用程序的开发, 提供了一个强大的、安全的执行环境, 支持多语言, 并简化了应用程序的部署和管理。公共语言运行库也称为“托管环境”, 在这个托管环境中将自动提供诸如垃圾回收和安全性等通用服务。

用 C# 编写的源代码被编译为一种符合 CLI 规范的中间语言 (IL)。IL 代码与资源 (如位图和字符串) 一起作为一种称为程序集的可执行文件存储在磁盘上, 通常具有的扩展名为 .exe 或 .dll。程序集包含清单, 它提供有关程序集的类型、版本、区域性和安全要求等信息。

执行 C# 程序时, 程序集将加载到 CLR 中, 这可能会根据清单中的信息执行不同的操作。然后, 如果符合安全要求, CLR 就会执行实时 (JIT) 编译以将 IL 代码转换为本机机器指令。CLR 还提供与自动垃圾回收、异常处理和资源管理有关的其他服务。由 CLR 执行的代码有时称为“托管代码”, 它与编译为面向特定系统的本机机器语言的“非托管代码”相对应。图 1-1 阐释了 C# 源代码文件、.NET Framework 类库、程序集和 CLR 的编译时与运行时的关系。

语言互操作性是 .NET Framework 的一项主要功能。因为由 C# 编译器生成的 IL 代码符合公共类型规范 (CTS), 因此从 C# 生成的 IL 代码可以与从 Visual Basic、Visual C++、Visual J# 的 .NET 版本或者其他 20 多种符合 CTS 的语言中的任何一种生成的代码进行交互。单一程序集可能包含用不同 .NET 语言编写的多个模块, 并且类型可以相互引用, 就像它们是用同一种语言编写的。

运行库还提高了开发人员的工作效率。例如, 程序员可以用他们选择的开发语言编写应用程序, 却仍能充分利用其他开发人员用其他语言编写的运行库、类库和组件。任何选择以运行库为目标的编译器供应商都可以这样做。以 .NET Framework 为目标的语言编译器使得用该语言编写的现有代码可以使用 .NET Framework 的功能, 这大大减轻了现有应用程序的迁移过程的工作负担。

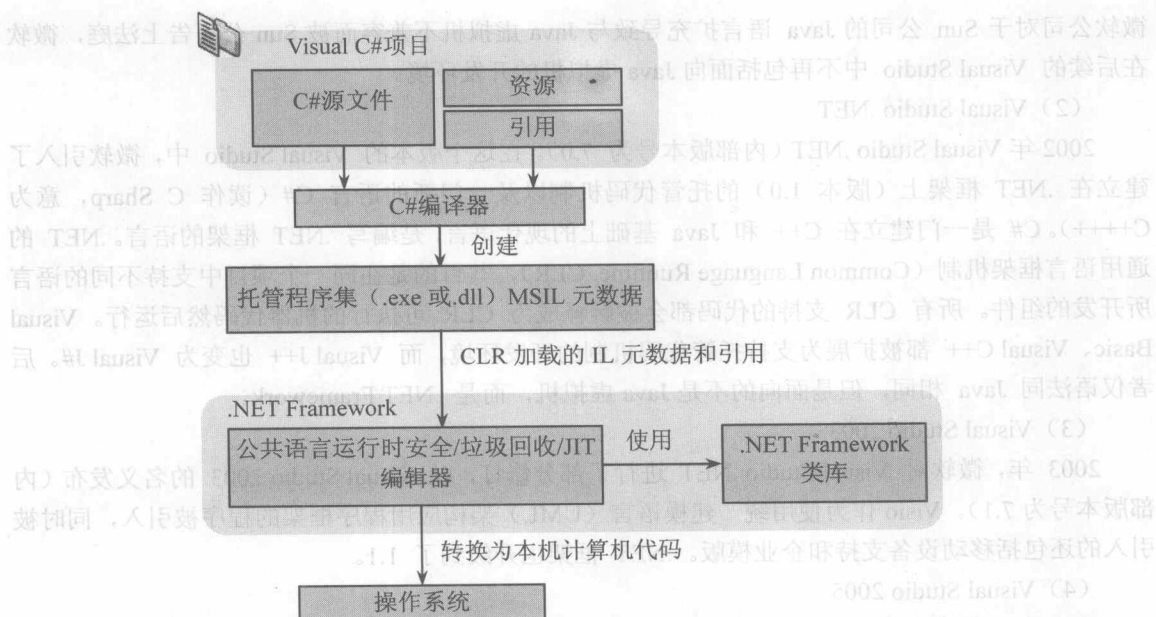


图 1-1 关系图

公共语言运行库负责运行时服务，如语言集成、强制安全，以及内存、进程和线程管理。除此之外，它还在开发时期承担如生命周期管理、强类型命名、跨语言异常处理以及动态绑定之类的角色，以减少开发人员将事务逻辑转成可重用组件必须写的代码数量。

公共语言运行库为开发人员构建不同类型的应用程序提供了可靠的基础，让设计含有跨语言对象的组件与应用程序变得更加容易。不同语言写成的对象可以互相通信，它们的行为可以被紧密集成。

2. .NET Framework 类库

在早期的开发中，各种应用开发人员使用各自平台所提供的工具类库，开发适用于不同平台的应用时，则要求开发人员掌握多种类库的使用方法，造成大量的资源浪费，也降低了开发人员的效率。

.NET 框架提供了丰富的接口集合，以及抽象与非抽象类。开发人员可以原封不动使用非抽象类，或者在许多情况下，派生出自定义的类。要使用接口的功能，开发人员既可以创建一个实现接口的类，也可以从某个实现该接口的 .NET 框架类中派生出类。

曾经难于实现，或是需要第三方组件支持的应用程序特性，如今使用 .NET 框架后，可以通过少量代码得以实现。.NET Framework 还包含一个由 4000 多个类组成的内容详尽的库，这些类被组织为命名空间，为从文件输入和输出、字符串操作、XML 分析到 Windows 窗体控件的所有内容提供了各种有用的功能。

1.2 Visual Studio 2008 简介

1. 历史沿革

(1) Visual Studio 6.0

1998 年，微软公司发布了 Visual Studio 6.0。所有开发语言的开发环境版本均升至 6.0。由于

微软公司对于 Sun 公司的 Java 语言扩充导致与 Java 虚拟机不兼容而被 Sun 公司告上法庭, 微软在后续的 Visual Studio 中不再包括面向 Java 虚拟机的开发环境。

(2) Visual Studio .NET

2002 年 Visual Studio .NET (内部版本号为 7.0)。在这个版本的 Visual Studio 中, 微软引入了建立在 .NET 框架上 (版本 1.0) 的托管代码机制以及一门新的语言 C# (读作 C Sharp, 意为 C++++)。C# 是一门建立在 C++ 和 Java 基础上的现代语言, 是编写 .NET 框架的语言。.NET 的通用语言框架机制 (Common Language Runtime, CLR), 其目的是在同一个项目中支持不同的语言所开发的组件。所有 CLR 支持的代码都会被解释成为 CLR 可执行的机器代码然后运行。Visual Basic、Visual C++ 都被扩展为支持托管代码机制的开发环境, 而 Visual J++ 也变为 Visual J#。后者仅语法同 Java 相同, 但是面向的不是 Java 虚拟机, 而是 .NET Framework。

(3) Visual Studio 2003

2003 年, 微软对 Visual Studio .NET 进行了部分修订, 以 Visual Studio 2003 的名义发布 (内部版本号为 7.1)。Visio 作为使用统一建模语言 (UML) 架构应用程序框架的程序被引入, 同时被引入的还包括移动设备支持和企业模版。.NET 框架也升级到了 1.1。

(4) Visual Studio 2005

2005 年, 微软发布了 Visual Studio 2005。.NET 字眼从各种语言的名字中被抹去, 但是这个版本的 Visual Studio 仍然还是面向 .NET 框架的 (版本 2.0)。它同时也能开发跨平台的应用程序, 如开发使用微软操作系统的手机的程序等。总体来说是一个非常庞大的软件, 甚至包含代码测试功能。这个版本的 Visual Studio 包含有众多版本, 分别面向不同的开发角色。同时还永久提供免费的 Visual Studio Express 版本。

(5) Visual Studio 2008

Microsoft Visual Studio 2008 使开发人员能够快速创建高质量、用户体验丰富而又紧密联系的应用程序, 充分展示了 Microsoft 开发智能客户端应用程序的构想。借助 Visual Studio 2008, 采集和分析信息将变得更为简单便捷, 业务决策也会因此变得更为有效。任何规模的组织都可以使用 Visual Studio 2008 快速创建能够利用 Windows Vista™和 Office System 2007 的更安全、更易于管理并且更可靠的应用程序。

Visual Studio 2008 在三个方面为开发人员提供了关键改进:

- 快速的应用程序开发;
- 高效的团队协作;
- 突破性的用户体验。

如果你有一定的编程基础, 希望了解 Visual C# 2008 和以前版本相比较的重大更改, 请参考网址 <http://msdn.microsoft.com/zh-cn/library/cc713578.aspx>。

2. Visual Studio 2008 的安装

(1) 支持的操作系统

- Microsoft Windows XP;
- Microsoft Windows Server 2003;
- Windows Vista。

(2) 硬件要求

最低要求: 1.6 GHz CPU、384 MB RAM、1024×768 显示器、5400 RPM 硬盘。

建议配置：2.2GHz 或速度更快的 CPU、1024MB 或更大容量的 RAM、1280×1024 显示器、7200RPM 或更高转速的硬盘。

在 Windows Vista 上：2.4GHz CPU、768MB RAM。

3. C# 语言简介

C# 是一种简洁、类型安全的面向对象的语言，开发人员可以使用它来构建在 .NET Framework 上运行的各种安全、可靠的应用程序。使用 C#，可以创建传统的 Windows 客户端应用程序、XML Web Services、分布式组件、客户端/服务器应用程序、数据库应用程序等。Visual C# 2008 提供了高级代码编辑器、方便的用户界面设计器、集成调试器和许多其他工具，可以更容易在 C#语言 3.0 版和 .NET Framework 3.5 版的基础上开发应用程序。

C# 语法表现力强，而且简单易学。C#的大括号语法使任何熟悉 C、C++或 Java 的人都可以立即上手。了解上述任何一种语言的开发人员通常在很短的时间内就可以开始使用 C#高效地进行工作。C# 语法简化了 C++的诸多复杂性，并提供了很多强大的功能，例如可为 null 的值类型、枚举、委托、lambda 表达式和直接内存访问，这些都是 Java 所不具备的。C#支持泛型方法和类型，从而提供了更出色的类型安全和性能。C#还提供了迭代器，允许集合类的实施者定义自定义的迭代行为，以便容易被客户端代码使用。在 C# 3.0 中，语言集成查询（LINQ）表达式使强类型查询成为了一流的语言构造。

作为一种面向对象的语言，C#支持封装、继承和多态性的概念。所有的变量和方法，包括 Main 方法（应用程序的入口点）都封装在类定义中。类只能直接从一个父类继承，但它可以实现任意数量的接口。重写父类中的虚方法的各种方法要求 override 关键字作为一种避免意外重定义的方式。在 C#中，结构类似于一个轻量类；它是一种堆栈分配的类型，可以实现接口，但不支持继承。

除了这些基本的面向对象的原理之外，C#还通过几种创新的语言构造，简化了软件组件的开发，这些结构包括：

- 封装的方法签名（称为“委托”），它实现了类型安全的事件通知。
- 属性（Property），充当私有成员变量的访问器。
- 字段（Field），是类或结构的数据成员。
- 内联 XML 文档注释。
- 语言集成查询（LINQ），提供了跨各种数据源的内置查询功能。

在 C# 中，如果必须与其他 Windows 软件（如 COM 对象或本机 Win32 DLL）交互，则可以通过一个称为“互操作”的过程来实现。互操作使 C# 程序能够完成本机 C++应用程序可以完成的几乎任何任务。在直接内存访问必不可少的情况下，C# 甚至支持指针和“不安全”代码的概念。

C# 的生成过程比 C 和 C++简单，比 Java 更为灵活。没有单独的头文件，也不要求按照特定顺序声明方法和类型。C# 源文件可以定义任意数量的类、结构、接口和事件。

如果你有 Java 编程基础，想了解 C#和 Java 编程语言之间的相似点和差异请参考网址：<http://msdn.microsoft.com/zh-cn/library/ms228602.aspx>。

如果你是一位 C++ 程序员，想了解 C#和 Java 编程语言之间的相似点和差异请参考网址：<http://msdn.microsoft.com/zh-cn/library/yyaad03b.aspx>。

1.3 Visual C#.NET 集成开发环境简介

1. 起始页

单击“开始”→“所有程序”→Microsoft Visual Studio 2008→Microsoft Visual Studio 2008 命令，启动 VS 2008（Microsoft Visual Studio 2008 的缩写），在默认情况下会显示图 1-2 所示的起始页。若起始页关闭，可以通过单击“视图”→“其他窗口”→“起始页”命令打开起始页。

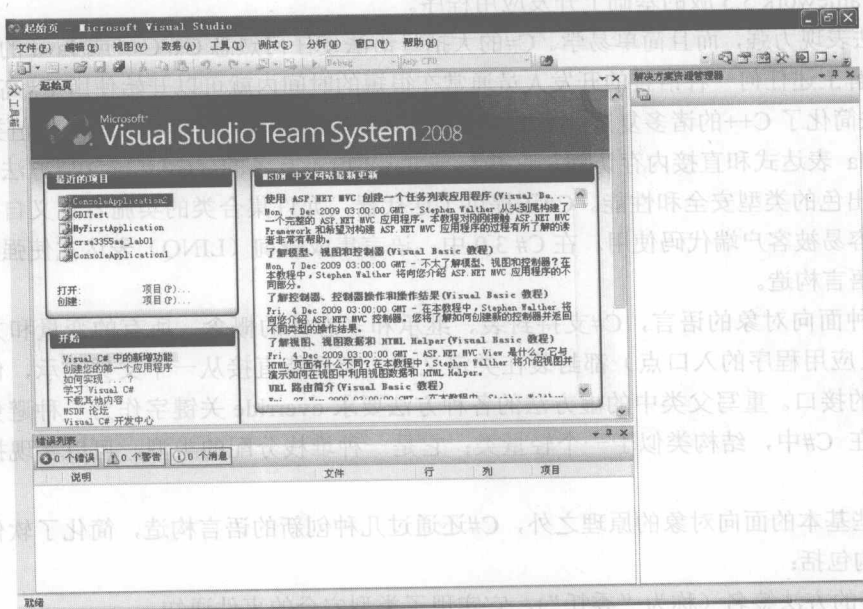


图 1-2 Visual Studio 2008 起始页

“最近的项目”显示最近打开过的项目列表，可以更方便地创建新项目和打开现有项目。

“开始”显示一个列表，其中包括帮助主题、网站、技术文章以及旨在提高工作效率的一些有用链接，初学者可以作为参考。该主题列表根据应用的设置的不同而变化。

“Visual Studio 标题新闻”显示微软公司的产品和事件信息的链接。

“MSDN 中文网站最新更新”在默认情况下，显示应用的设置指定的 RSS 源中的文章列表。可以通过菜单“工具”→“选项”→“环境”→“启动”→“起始页新闻频道”中输入新闻 RSS 源的 URL。

2. 开发环境

当打开或者新建一个项目后，Visual Studio 2008 的典型开发环境如图 1-3 所示。由于 Visual Studio 2008 的开发环境布局可以定制，如果你机器上的 Visual Studio 2008 被其他人使用过，你看到的界面有可能会与图 1-3 不同。

Visual C# 集成开发环境 (IDE) 是一种通过常用用户界面公开的开发工具的集合。有些工具是与其他 Visual Studio 语言共享的，还有一些工具（如 C# 编译器）是 Visual C# 特有的。本节介绍如何在使用 IDE 时针对开发过程的各个阶段使用最重要的 Visual C# 工具。

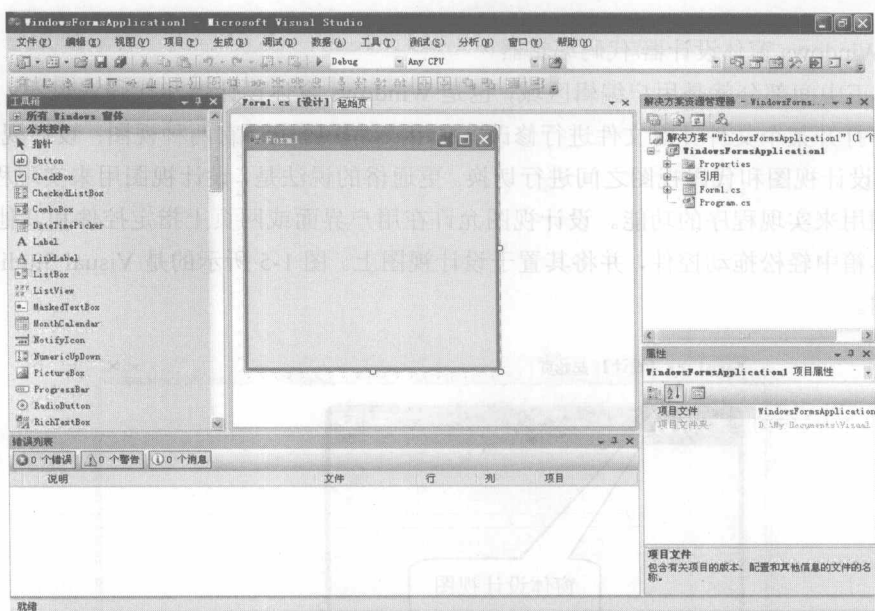


图 1-3 Visual Studio 2008 开发环境

以下是 Visual C# 中最重要的工具和窗口。

- 代码编辑器，用于编写源代码。
- C# 编译器，用于将 C# 源代码转换为可执行程序。
- Visual Studio 调试器，用于对程序进行测试。
- 工具箱，用于使用鼠标快速开发用户界面。
- 解决方案资源管理器，用于查看和管理项目文件和设置。
- 类视图，用于根据类型（而不是文件）在源代码中导航。
- “属性”窗口，用于配置用户界面中控件的属性和事件。
- 对象浏览器，用于查看动态链接库（包括 .NET Framework 程序集和 COM 对象）中可用的方法和类。

大多数工具的窗口可从“视图”菜单打开，仅介绍初学者需要使用的 5 个窗口，如图 1-4 所示。

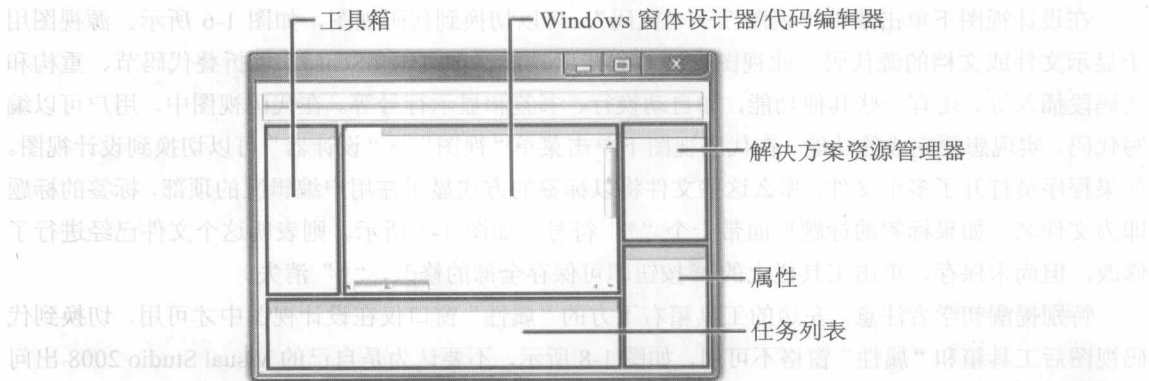


图 1-4 Visual C# 集成开发环境 (IDE) 示意图