

教材+教案+授课资源+考试系统+题库+教学辅助案例
一站式IT就业培训系列教程

C# 程序设计基础入门教程

传智播客高教产品研发部 编著



◎ 原版图书配套DVD教学光盘，内含教学视频+案例源码

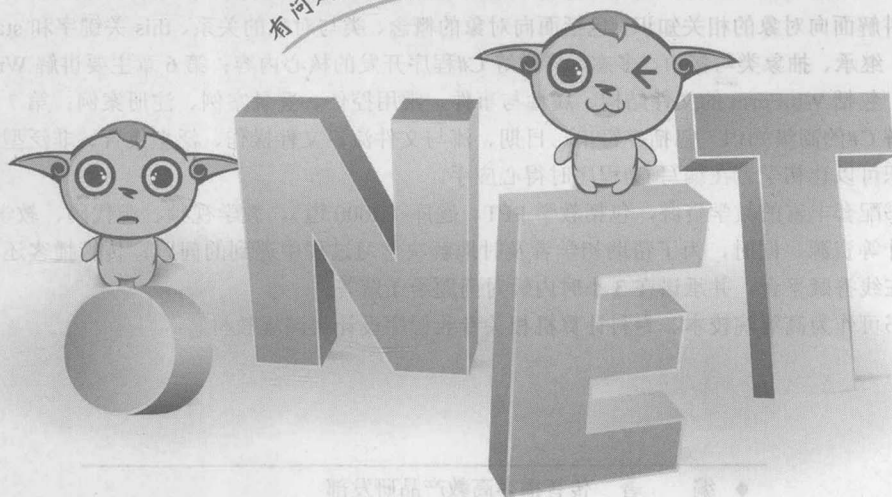
 人民邮电出版社
POSTS & TELECOM PRESS

C# 程序设计基础入门教程

传智播客高教产品研发部 编著

要 目 录

有问题，就找问答精灵！



◆ 第1章 初识C# 1

◆ 第2章 数据类型 11

◆ 第3章 运算符 21

◆ 第4章 控制语句 31

◆ 第5章 数组 41

◆ 第6章 字符串 51

◆ 第7章 集合 61

◆ 第8章 异常 71

◆ 第9章 线程 81

◆ 第10章 网络 91

◆ 附录A 常用API 101

◆ 附录B 常用API 111

人民邮电出版社

北京

图书在版编目 (CIP) 数据

C#程序设计基础入门教程 / 传智播客高教产品研发
部编著. — 北京: 人民邮电出版社, 2014. 9
ISBN 978-7-115-35624-6

I. ①C… II. ①传… III. ①C语言—程序设计—教材
IV. ①TP312

中国版本图书馆CIP数据核字(2014)第148209号

内 容 提 要

C#作为微软的旗舰编程语言,深受程序员喜爱,是编写高效应用程序的首选语言。《C#程序设计基础入门教程》就是面向零基础的学生推出的一本入门书籍,本书从初学者角度,以通俗易懂的语言、丰富多彩的实例,详细讲解了使用C#语言进行程序开发时应该掌握的各种技术。

本书共9章:第1章至3章主要讲解C#的基础知识,包括.NET Framework、Visual Studio开发环境的搭建、变量与常量、表达式与运算符、选择结构语句、流程控制语句、数组、方法;第4章至第5章主要讲解面向对象的相关知识,包括面向对象的概念、类与对象的关系、this关键字和static关键字、索引器、继承、抽象类与接口、多态、异常等C#程序开发的核心内容;第6章主要讲解WinForm的基础知识,包括WinForm的文件结构、属性与事件、常用控件、登录案例、注册案例;第7章至第9章主要讲解C#的高级知识,包括字符串、日期、流与文件流、文件操作、泛型集合、非泛型集合,掌握这些知识可以让初学者在编写C#程序时得心应手。

本书配套丰富的教学资源,包括教学PPT、题库(2000道)、教学视频、源代码、教学补充案例、教学设计等资源。同时,为了帮助初学者及时地解决学习过程中遇到的问题,传智播客还专门提供了免费的在线答疑平台,并承诺在3小时内针对问题给予解答。

本书可作为高等院校本、专科计算机相关专业程序设计类课程教材。

◆ 编 著 传智播客高教产品研发部

责任编辑 范博涛

责任印制 杨林杰

◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号

邮编 100164 电子邮件 315@ptpress.com.cn

网址 <http://www.ptpress.com.cn>

三河市中晟雅豪印务有限公司印刷

◆ 开本: 787×1092 1/16

印张: 17.25

2014年9月第1版

字数: 430千字

2014年9月河北第1次印刷

定价: 45.00元(附光盘)

读者服务热线: (010)81055256 印装质量热线: (010)81055316

反盗版热线: (010)81055315

序言

为什么出这本书

数年前，传智播客团队是一群默默工作在 IT 岗位的程序员，现如今，为了同一个梦想，我们聚集在一起，为“改变中国的 IT 教育”而全力以赴。为什么会有这样的梦想，大家先来听一听 IT 毕业生的心声：

“我们专业的课程太枯燥了，太多理论方面的课程，并且这些理论听起来都是很难理解的，同学们学完了都是一头雾水。”

“终于毕业了，学了几年的计算机课程，我们班上大部分同学都不知道能干什么，感觉好像什么都学了，但却什么都不会干！”

“实习单位根本就不让我们干活，最多只是让我们阅读代码、修改文档，说我们什么都不会，不敢让我们参与实际工作。想想自己辛苦学了几年连实习的要求都达不到，心里很不是滋味。”

“面试的时候，用人单位问能做什么，做过什么项目，我们回答不上来，面试过程中问到的很多问题，我们连听都没有听过。”

.....

这些心声并不是个别毕业生所表达的，而是中国高校的一种普遍现象。高校的 IT 教育与企业的真实需求存在严重脱节，如果高校的相关课程仍不进行更新的话，毕业生将继续面临毕业即失业的困境。出现这样的困境主要原因在于教材更新速度缓慢，导致学生所学的知识与现在的社会需求严重不符，学校无法向学生传授企业需要的专业技能。面对高校所存在的问题，国务院近期提出了提高高等教育质量的改革要求，很庆幸，我们所做的事情就是配合高校完成教学改革。

传智播客作为一家专业的 IT 培训机构，一直将“改变中国的 IT 教育”作为自己的事业，并为此拼搏了 9 年。在这期间，传智播客默默耕耘，陆续出版了计算机书籍十几本、教学视频几十套、发表各类技术论文几百篇，直接培养的软件工程师就有 2 万多名，被传智播客影响的在校学生更是多达数百万人。毫不夸张地说，传智播客已经为 IT 学子开辟了一条全新的求知之路。

9 年的沉淀让传智播客拥有了目前国内最好的 IT 培训课程。如今，传智播客决定把这套精品课程全部公开，印刷成书并输送至高校，解决眼下高校最急待解决的课程陈旧问题。另外，面向高校计算机专业的老师，传智播客还将提供免费的培训服务，让高校老师可以通过培训提高授课技能，将最专业、最实用的技能传授给学生。为了让广大师生在学习传智播客课程时能达到最好的效果，公司投入巨额资金，用于为高校师生提供以下学习配套资源与服务：

1. 本书非常荣幸地落户于在线学习网站【“博学谷”：<http://www.boxuegu.com>】。大家都知道：“学习 IT 的人都在博学谷”，难道您真不知道？

2. 在学习本书过程中，遇到任何问题，请找问答精灵，我们承诺工作时间 3 小时内解答您的问题。问答精灵的官方网址为：<http://ask.boxuegu.com>。读者可以通过扫描下面的二维码，下载问答精灵移动客户端或关注问答精灵微信公众平台。学习 IT 有问题就找问答精灵。



问答精灵移动客户端



问答精灵微信公众平台

3. 我们为本书录制了全程教学视频,通过学习视频您可以更容易理解本书的知识,视频下载网址: <http://dvd.boxuegu.com/net>。

4. 心中有疑问,想找老师面对面解惑?每周六晚上8点至10点,“面对面”讲堂,我们与您不见不散,网络公开课网址: <http://openclass.boxuegu.com/net>。

5. 学习需要氛围,在校园快点组织起您的学习社团吧,只要申请通过,我们将免费给予社团各种支持,包括教材、视频教程、技术参考资料、技术面试资料、技术讲座、实习岗位等,优秀社团还可获得现金奖励。

学习社团报名网址: <http://club.boxuegu.com>。

6. 本书配套源代码,下载网址: <http://book.boxuegu.com/net>。

7. 针对高校教学,传智播客用近9年的教育培训经验,精心设计了“教材+教案+授课资源+考试系统+题库+教学辅助案例”一站式IT就业培训系列教程,方便老师进行教学,能够有效提高老师日常教学的效率。

如需索要配套教学资源,请关注博学谷高校老师俱乐部,扫描下方二维码关注微信公众平台。

希望通过我们的努力,在不久的将来,高校能够更多地培养出符合企业所需的实用型人才,IT学子们不再为就业而迷惘!

关于本教材

作为一种技术的入门教程最重要也最难的一件事情就是要将一些非常复杂、难以理解的问题和思想简单化,让初学者能够轻松理解并快速掌握。本教材对每个知识点都进行了深入的分析,并针对每个知识点精心设计了相关案例,然后模拟这些知识点在实际工作中的运用,真正做到了知识的由浅入深、由易到难。为确保教材通俗易懂,在教材编写的过程中,我们还让600多名初学者参与到了教材试读中,对初学者反馈上来的难懂地方均作了一一修改。因此本书将是您接触到的技术书籍中最通俗易懂的一本。

本教材共分为9个章节,接下来分别对每个章节进行简单的介绍,具体如下。

第1章主要介绍C#程序开发的平台及其语言、C#开发环境的搭建,并且带领初学者开发一个最简单的HelloWorld程序,亲身体验C#语言的开发、运行过程。通过本章的学习,初学者能够对C#语言有一个整体的认识,并且可以熟练使用VS工具开发一个简单的程序。

第2、3章主要介绍C#编程基础知识,包括数据类型、常量、变量、方法、数组及其程序的调试。只有熟练掌握了C#中的基础知识,才能得心应手地学习后续章节。

第4、5章主要讲解面向对象的基本理论及思想,包括类与对象、构造方法、索引器、对象初始化器、继承、多态、抽象类、接口、异常、命名空间与程序集等。C#是一种面向对象的语言,深入理解面向对象的编程思想在学习C#语言的过程中尤为重要。

第6章主要介绍WinForm基础的相关知识,包括WinForm项目中的文件结构,常用



高校老师俱乐部

WinForm 控件的使用等。由于本章所讲解的内容侧重于实践操作,因此,建议初学者通过动手操作来加深对本章内容的理解。

第 7 章主要介绍 C# 中的常用类,包括 String 类、StringBuilder 类、DateTime 和 Random 类。熟练掌握这些常用类的使用,可以使初学者更好地开发 C# 程序。

第 8 章主要介绍 C# 中的流及其文件操作,包括文件流、File 类、FileInfo 类、Path 类、对象的序列化与反序列化等。文件是存储数据最常用的一种媒介,只有熟练掌握了文件的相关操作,才能更好地完成数据的交互。

第 9 章主要介绍集合,包括 ArrayList 集合、Hashtable 集合、泛型集合等。与数组相比,使用集合存储数据更加灵活,并且可以包含更高级的功能。

在学习本教材时,首先要做到对知识点理解透彻,其次一定要亲自动手去练习教材中所提供的案例,因为在学习软件编程的过程中动手实践是非常重要的。对于一些非常难以理解的知识点也可以选择通过案例的练习来学习,如果实在无法理解教材中所讲解的知识,建议初学者不要纠结于某一个知识点,可以先往后学习。通常来讲,看了后面一两个小节的内容后再回来学习之前不懂的知识点一般就都能理解了。

致谢

本教材的编写和整理工作由传智播客教育科技有限公司高教产品研发部完成,主要参与人员有徐文海、王春生、贡宗新、李允晶、高美云、陈欢、黄云、王龙、马丹,研发小组全体成员在这近一年的编写过程中付出了很多辛勤的汗水。另外,传智播客讲师马伦、牛亮亮、杨洪波、刘岐、汪磊、赵剑宇、王承伟、刘晓曲、栾博、许瑞航等人也参与了教材的部分编写和修订工作,除此之外,还有传智播客 600 多名学员也参与到了教材的试读工作中,他们站在初学者的角度对教材提供了许多宝贵的修改意见,在此一并表示衷心的感谢。

意见反馈

尽管我们尽了最大的努力,但教材中难免会有不妥之处,欢迎各界专家和读者朋友们来信来涵给予宝贵意见,我们将不胜感激。您在阅读本书时,如发现任何问题或有不认同之处可以通过电子邮件与我们取得联系。

请发送电子邮件至: itcast_book@vip.sina.com

传智播客教育科技有限公司 高教产品研发部

2014 年 6 月于北京

目 录 CONTENTS

第 1 章 C#开发入门 1

1.1 .NET 基础知识	1	1.2.3 Visual Studio 启动过程	6
1.1.1 .NET Framework	1	1.2.4 Visual Studio 主界面	7
1.1.2 C#语言	2	1.3 编写 HelloWorld 程序	8
1.2 开发环境搭建	3	1.4 运行原理	11
1.2.1 Visual Studio 介绍	3	1.5 本章小结	12
1.2.2 Visual Studio 安装	4	1.6 习题	12

第 2 章 C#编程基础（上） 13

2.1 C#的基本语法	13	2.3.4 逻辑运算符	29
2.1.1 C#代码的基本格式	13	2.3.5 位运算符	31
2.1.2 C#中的注释	14	2.3.6 运算符的优先级	32
2.1.3 C#中的标识符	15	2.4 选择结构语句	33
2.1.4 C#中的关键字	15	2.4.1 if 条件语句	34
2.1.5 C#中的常量	16	2.4.2 switch 条件语句	39
2.2 C#中的变量	19	2.5 循环结构语句	42
2.2.1 变量的定义	19	2.5.1 while 循环语句	43
2.2.2 变量的数据类型	20	2.5.2 do...while 循环语句	44
2.2.3 变量的类型转换	22	2.5.3 for 循环语句	45
2.2.4 变量的作用域	25	2.5.4 跳转语句（break、goto、continue）	47
2.3 C#中的运算符	26	2.5.5 循环嵌套	49
2.3.1 算术运算符	26	2.6 本章小结	51
2.3.2 赋值运算符	28	2.7 习题	51
2.3.3 比较运算符	28		

第 3 章 C#编程基础（下） 55

3.1 方法	55	3.3.3 多维数组	76
3.1.1 什么是方法	55	3.4 程序调试	77
3.1.2 方法的重载	58	3.4.1 设置断点	77
3.2 方法的高级应用	61	3.4.2 单步调试	78
3.2.1 递归调用	61	3.4.3 观察变量	79
3.2.2 高级参数	62	3.4.4 条件断点	81
3.3 数组	67	3.5 本章小结	83
3.3.1 数组的定义	67	3.6 习题	83
3.3.2 数组的常见操作	72		

第4章 面向对象基础 87

4.1 面向对象的概念	87	4.8 static 关键字	108
4.2 类与对象	88	4.8.1 静态字段	108
4.2.1 类的定义	88	4.8.2 静态属性	109
4.2.2 对象的创建与使用	89	4.8.3 静态方法	110
4.2.3 类的设计	92	4.8.4 静态类	111
4.2.4 属性	93	4.8.5 静态构造方法	111
4.3 访问修饰符	96	4.8.6 单例模式	112
4.4 构造方法	96	4.9 嵌套类	114
4.4.1 构造方法的定义	96	4.10 匿名类	115
4.4.2 构造方法的重载	98	4.11 对象初始化器	116
4.5 this 关键字	101	4.12 本章小结	117
4.6 索引器	104	4.13 习题	117
4.7 垃圾回收	106		

第5章 面向对象高级 122

5.1 类的继承	122	5.3.5 Object 类	140
5.1.1 继承的概念	122	5.4 抽象类和接口	142
5.1.2 构造方法的执行过程	124	5.4.1 抽象类	142
5.1.3 隐藏基类方法	126	5.4.2 接口	143
5.1.4 装箱与拆箱	127	5.5 异常	146
5.2 sealed 关键字	128	5.5.1 什么是异常	146
5.2.1 sealed 关键字修饰类	128	5.5.2 try...catch 和 finally	148
5.2.2 sealed 关键字修饰方法	129	5.5.3 throw 关键字	151
5.3 多态	130	5.6 命名空间与程序集	152
5.3.1 重写父类方法	131	5.6.1 命名空间	152
5.3.2 多态的实现	132	5.6.2 程序集	153
5.3.3 base 关键字	133	5.7 本章小结	156
5.3.4 里氏转换原则	136	5.8 习题	156

第6章 WinForm 基础 162

6.1 WinForm 简介	162	6.2.3 ListBox 列表框、ComboBox 下拉 列表框	176
6.1.1 WinForm 项目的文件结构	162	6.2.4 GroupBox 容器	180
6.1.2 控件的常用属性与事件	165	6.2.5 TreeView 树视图控件	182
6.1.3 创建第一个 WinForm 应用程序	166	6.2.6 Timer 控件	185
6.2 WinForm 常用控件	170	6.2.7 ProgressBar 控件	187
6.2.1 Button 按钮、TextBox 文本框、Label 文本标签	171	6.3 本章小结	190
6.2.2 CheckBox 复选框、RadioButton 单选按钮	173	6.4 习题	190

第7章 C#常用类 194

7.1 字符串基本特征	194	7.3.2 StringBuilder 性能分析	208
7.1.1 字符串的不可变性	194	7.4 日期处理	209
7.1.2 字符串与字符数组	195	7.4.1 DateTime 类	209
7.2 String 类的常用方法	196	7.4.2 DateTime 类的常用属性	211
7.2.1 String 类的构造方法	196	7.4.3 DateTime 类的常用方法	212
7.2.2 String 类的静态方法	197	7.5 Random 类	214
7.2.3 String 类的实例方法	201	7.6 本章小结	217
7.3 高效的 StringBuilder	206	7.7 习题	217
7.3.1 StringBuilder 类	206		

第8章 文件操作 221

8.1 流和文件流	221	8.5.3 FileStream 类写入文件	231
8.2 System.IO 命名空间	222	8.5.4 文件复制	232
8.3 File 类和 FileInfo 类	222	8.6 StreamReader 类和 StreamWriter 类	234
8.3.1 File 类	222	8.6.1 StreamWriter 类	234
8.3.2 FileInfo 类	224	8.6.2 StreamReader 类	236
8.4 Directory 类和 DirectoryInfo 类	226	8.6.3 文件读写	238
8.4.1 Directory 类	226	8.7 Path 类	239
8.4.2 DirectoryInfo 类	227	8.8 BufferedStream 类	241
8.5 FileStream 类	229	8.9 序列化和反序列化	243
8.5.1 FileStream 类介绍	229	8.10 本章小结	245
8.5.2 FileStream 类读取文件	229	8.11 习题	245

第9章 集合 249

9.1 集合概述	249	9.3.1 List<T>泛型集合	259
9.2 非泛型集合	250	9.3.2 Dictionary<TKey, TValue> 泛型集合	261
9.2.1 ArrayList 集合	250	9.3.3 自定义泛型	262
9.2.2 foreach 循环	256	9.4 本章小结	263
9.2.3 Hashtable 集合	257	9.5 习题	264
9.3 泛型集合	259		

学习目标

- ◆ 认识.NET平台与C#语言
- ◆ 掌握C#开发环境的搭建
- ◆ 掌握HelloWorld程序的编写

对于初学者而言,学习任何一门语言都需要先认识它的运行平台。本书中所讲解的C#语言以及其编译和运行都依赖于.NET Framework平台。通常来讲,开发人员会将C#语言和.NET Framework平台一起使用,从而提高开发效率。本章作为本书的第1章,将针对.NET平台、C#语言以及开发环境的搭建进行详细的讲解。

1.1 .NET 基础知识

1.1.1 .NET Framework

.NET Framework 是 Microsoft 为开发应用程序而创建的一个平台。使用.NET Framework 可以创建桌面应用程序、Web 应用程序、Web 服务和其他各种类型的应用程序。它是一个全新的、集成的、面向对象的开发平台,它以公共语言运行时 (Common Language Runtime, 简写 CLR) 为基础,支持多种编程语言。现在.NET Framework 最常用的版本有.NET4.0 (Visual Studio 2010)、.NET4.5 (Visual Studio 2012) 和.NET4.5 (Visual Studio 2013) 等。.NET Framework 自身的功能随着版本的不断升级而越来越丰富,正展现出诱人的前景。

.NET Framework 主要由公共语言运行时 (CLR) 和.NET Framework 基础类库 (Base Class Library, 简称 BCL) 两部分组成。接下来通过一个图例来描述.NET Framework 的基本结构,如图 1-1

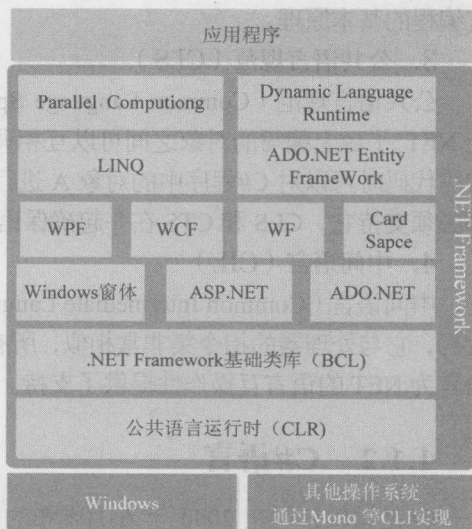


图 1-1 .NET 体系结构图

所示。

从图 1-1 可以看出, .NET Framework 位于操作系统与应用程序之间, 负责管理在 .NET Framework 上运行的各种应用程序。也就是说 .NET 应用程序不依赖于操作系统, 只依赖于 .NET Framework。 .NET Framework 底层是公共语言运行时 (CLR), 它可以支持多种编程语言。CLR 的上一层是 .NET Framework 的基础类库 (BCL), 它提供了 .NET 程序开发中常用的类库。再上一层是 Windows 窗体、ASP.NET、ADO.NET 等模块, 它们用于开发各种各样的应用程序, 如桌面应用程序、网络应用程序、企业级的应用程序等。其他的模块在 .NET 平台上广泛应用, 在后续的课程中会进行详细讲解。

通过前面的讲解, 初学者对 .NET Framework 有了简单的了解, 接下来对 .NET Framework 中的核心内容进行详细分析。

1. 公共语言运行时 (CLR)

.NET Framework 的核心是它的执行环境, 该环境称为公共语言运行时 (CLR) 或 .NET 运行时。公共语言运行时 (CLR) 主要负责管理 .NET 应用程序的编译、运行以及一些基础的服务, 它为 .NET 应用程序提供了一个虚拟的运行环境。同时 CLR 还负责为应用程序提供内存分配、线程管理以及垃圾回收等服务, 并且负责对代码实施安全检查, 以保证代码的正常运行。

通常将在 CLR 控制下运行的代码称为托管代码 (Managed Code), 它是体现 .NET Framework 优点的关键。如在开发 C/C++ 程序时, 需要手动管理和释放内存, 而在开发 C# 程序时, .NET Framework 会自动管理和释放内存, 当然鱼和熊掌不可兼得, 自动管理内存的特性会导致性能的降低。

2. 通用类型系统 (CTS)

CLR 有一个重要的组成部分, 即通用类型系统 (Common Type System, 简称 CTS)。各种主流语言都有自己的类型库, 但它们的类型系统大体相似, .NET 将各种不同编程语言的数据类型进行抽象, 就有了 CTS。CTS 为 .NET Framework 上的各种编程语言提供了支持, .NET Framework 上不同的编程语言通过编译后都转换为 CTS 类型。例如 Visual Basic .NET 中定义的 Integer 数据类型被转换成了 System.Int32 数据类型, C# 中的 int 类型也被转换成了 System.Int32 数据类型。这样, 不同语言的变量就可以相互交换信息了, 这就是 .NET Framework 支持混合语言编程的基本原理。

3. 公共语言规范 (CLS)

公共语言规范 (Common Language Specification, 简称 CLS) 是 CTS 的子集, 它的目的是让 .NET 平台上编写的对象之间可以互相调用, 实现语言的互操作性, 例如在 Visual Basic .NET 编写代码时可以对 C# 程序中的对象 A 进行操作。公共语言规范是一个标准集, .NET 的编译器都必须支持它, CLS 和 CTS 在一起确保语言的互操作性。

4. 中间语言 (CIL)

中间语言 (Common Intermediate Language, 简称 CIL) 是 C# 程序第一次编译后生成的托管代码, 它与处理器的指令集非常相似, 所有在 .NET 平台上的编程语言都会被编译为 CIL, 中间语言为 .NET 的语言互操作性提供了支持。

1.1.2 C# 语言

C# 是微软公司在 2000 年 6 月发布的一种全新的、简单的、安全的、面向对象的程序设计语言, 是专门为 .NET 的应用而开发的语言。它吸收了 C++、Visual Basic、Delphi、Java 等语言

的优点,体现了当今最新的程序设计技术的功能和精华。C#继承了C语言的语法风格,同时又继承了C++的面向对象特性。不同的是,C#的对象模型已经面向Internet进行了重新设计。

C#不再提供对指针类型的支持,使得程序不能随便访问内存地址空间,这使C#程序更加健壮。C#也不再支持多重继承,避免了以往类层次结构中由于多重继承带来的可怕后果。.NET Framework为C#提供了强大的、易用的、逻辑结构一致的程序设计环境。同时,公共语言运行时(CLR)为C#程序语言提供了一个托管的运行时环境,使程序比以往更加稳定、安全。

C#语言的特点可以概括为以下几点。

- 语言简洁。
- 语言的自由性。
- 保留了C++的强大功能。
- 强大的Web服务器控件。
- 支持跨平台。
- 与XML相融合。

需要注意的是,C#就其本身而言只是一种语言,尽管它是用于生成面向.NET Framework环境的代码,但它本身不是.NET的一部分。

1.2 开发环境搭建

在使用C#语言开发程序之前,首先要在系统中搭建开发环境,本书将使用Visual Studio开发工具来作为开发环境,该工具可以使程序开发变得更加简单。本节重点讲解的就是Visual Studio 2013开发工具的安装过程和使用。

1.2.1 Visual Studio 介绍

Microsoft Visual Studio(简称VS)是美国微软公司的开发工具包系列产品。VS是一个基本完整的开发工具集,它包括了整个软件生命周期所需要的大部分工具,如UML工具、代码管控工具、集成开发环境(IDE)等。使用VS编写的代码适用于微软支持的所有平台,包括Microsoft Windows、Windows Mobile、Windows CE、.NET Framework、.NET Compact Framework等。

Visual Studio是目前最流行的Windows平台应用程序的集成开发环境,最新版本为Visual Studio 2013,它基于.NET Framework 4.5.1开发。接下来简单介绍常用的Visual Studio集成开发环境。

1. Visual Studio 2010

Visual Studio 2010集成了ASP.NET MVC 4,全面支持移动和HTML5,并且期待已久的工作流回来了,更棒的是,它的设计器已经支持C#表达式(之前只能用VB.NET)。

2. Visual Studio 2012

VS2012支持.NET 4.5,和.NET 4.0相比,.NET 4.5更多的是完善和改进。.NET 4.5也是Windows RT被提出来的首个框架库,.NET获得了和Windows API同等的待遇。Visual Studio 2012对系统资源的消耗并不大,不过需要Windows 7/8的支持。

3. Visual Studio 2013

微软打破了 Visual Studio 两年升级一次的传统, Visual Studio 2012 发布还不足一年, 微软就发布了 Visual Studio 2013。Visual Studio 2013 新增了代码信息指示 (Code Information Indicators)、团队工作室 (Team Room)、身份识别、.NET 内存转储分析仪、敏捷开发项目模板、Git 支持以及更强大的单元测试支持。

1.2.2 Visual Studio 安装

上一节中主要介绍了 VS 的开发环境和版本, 本书以 Visual Studio 2013 为例讲解如何安装 VS 开发环境, 通过使用 VS 开发工具来进一步学习 .NET。具体安装步骤如下。

1. 开始安装

从微软的官网下载 VS2013_RTM_ULT_CHS.iso 镜像文件, 在本地可以直接解压或者通过虚拟光驱来进行安装, 解压后以管理员身份运行安装程序, 此时显示 Visual Studio 界面, 如图 1-2 所示。

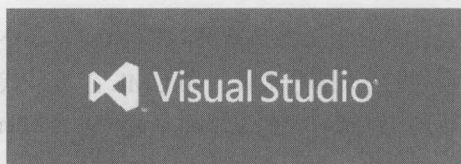


图 1-2 Visual Studio 界面

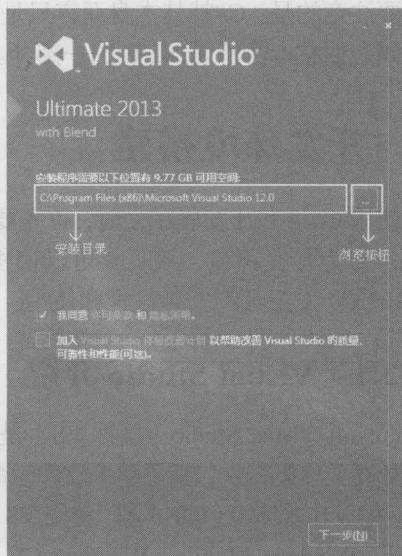


图 1-3 设置路径界面

图 1-2 所示的开始安装界面会停留一会儿, 然后便会进入设置路径界面, 如图 1-3 所示。

在图 1-3 中, 程序的安装目录默认为 C:\Program Files(x86)\Microsoft Visual Studio 12.0, 点击安装目录后的浏览按钮, 可以把 Visual Studio 开发工具安装到选定的目录, 本书使用默认路径进行安装, 选中【我同意许可条款和隐私策略】选项, 开始安装程序。关于【加入 Visual Studio 体验改善计划及帮助改善 Visual Studio 的质量、可靠性和性能 (可选)】选项, 用户可以根据自己的需求来选择。

2. 选择安装组件

点击图 1-3 中的【下一步】按钮, 进入安装组件选择界面, 如图 1-4 所示。

如图 1-4 所示, 界面中有一系列的安装组件, 开发人员可以根据自己的需求来选择所要安装的组件, 接下来对各个组件进行简单介绍。

- **Blend for Visual Studio**: 提供了准确的设计图面和工具来帮助用户设计和生成美观精致的用户界面，可以直观地创建和编辑 Microsoft Windows 应用程序。
- **LightSwitch**: 可以快速构建业务应用程序，让开发人员能够集中精力处理业务逻辑。
- 用于 C++ 的 **Microsoft 基础类**: 常用的 C++ 的基础类库。
- **Microsoft Office 开发人员工具**: 方便开发与 Office 相关的应用程序。
- **Microsoft SQL Server Data Tools**: 为数据库开发人员提供了一个集成的环境，让它们可以在 Visual Studio 内为任何 SQL Server 平台（无论是内部还是外部）执行所有数据库的设计工作。
- **Microsoft Web 开发人员工具**: 方便 Web 开发人员开发 Web 相关的应用程序。
- **Silverlight 开发工具包**: 提供了灵活的编程模型，可以方便地集成到现有的网络应用程序中。

3. 等待安装

在图 1-4 所示的界面中选择【全选】按钮，将所有组件全部选中，然后点击【安装】按钮，程序就会自动进行安装，如图 1-5 所示。

4. 安装完成

Visual Studio 安装完成后，会看到安装成功界面，如图 1-6 所示。

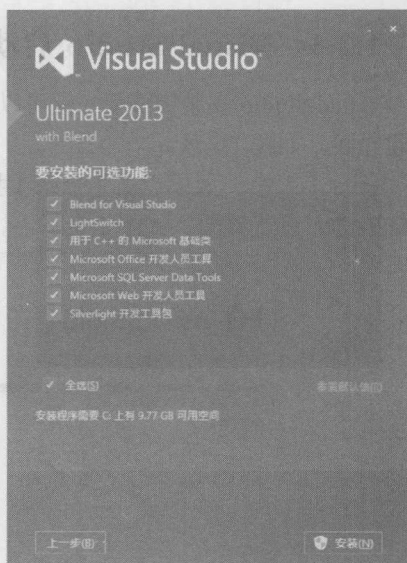


图 1-4 组件选择界面

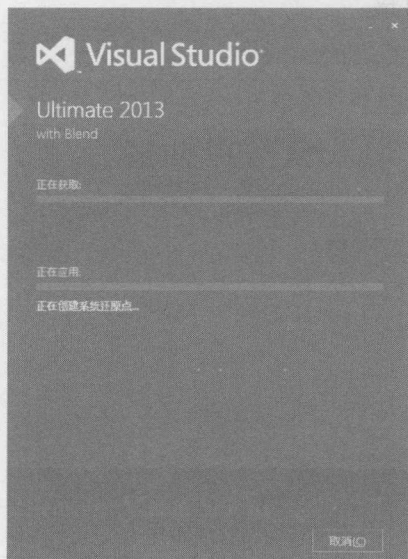


图 1-5 安装界面

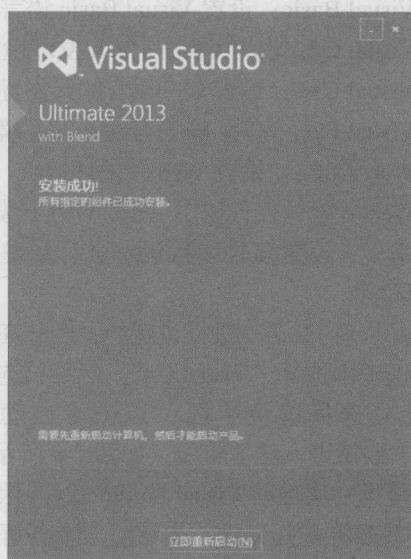


图 1-6 安装成功界面

1.2.3 Visual Studio 启动过程

Visual Studio 安装完成后, 就可以启动 Visual Studio 软件了, 首次启动 Visual Studio 会显示欢迎界面, 如图 1-7 所示。

在图 1-7 中, 如果点击【登录】按钮可以链接到在线开发者服务。在此点击“以后再说”超链接进入设置界面, 如图 1-8 所示。



图 1-7 欢迎界面

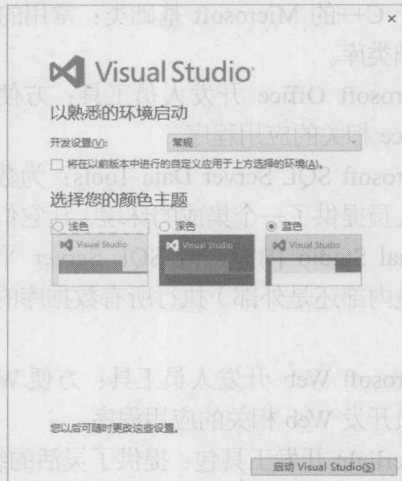


图 1-8 设置界面

在图 1-8 所示的设置界面中, 可以对 Visual Studio 环境进行“开发设置”, 让开发者选择熟悉的编程语言进行程序开发, 该选项提供了多种开发语言环境, 具体如下。

- JavaScript: 设置 JavaScript 语言的开发环境。
- SQL Server: 设置 SQL Server 数据库的开发环境。
- Visual Basic: 设置 Visual Basic 语言的开发环境。
- Visual C#: 设置 C# 语言的开发环境。
- Visual C++: 设置 C++ 语言的开发环境。
- Visual F#: 设置 F# 语言的开发环境。
- Web 开发: 设置 Web 开发环境。
- Web 开发 (仅代码): 设置 Web 开发环境。
- 常规: 设置多种编程语言进行开发的开发环境。

本书是针对 C# 语言进行讲解的, 因此在界面的开发设置选项中选择“Visual C#”。然后再选择颜色主题, 此处选择蓝色主题, 点击【启动 Visual Studio(S)】按钮进入准备阶段, 如图 1-9 所示。

由于是第一次启动 Visual Studio 软件, 因此这个过程可能需要等待几分钟, 当该阶段准备好了以后就会进入 Visual Studio 的起始页, 如图 1-10 所示。



图 1-9 准备阶段

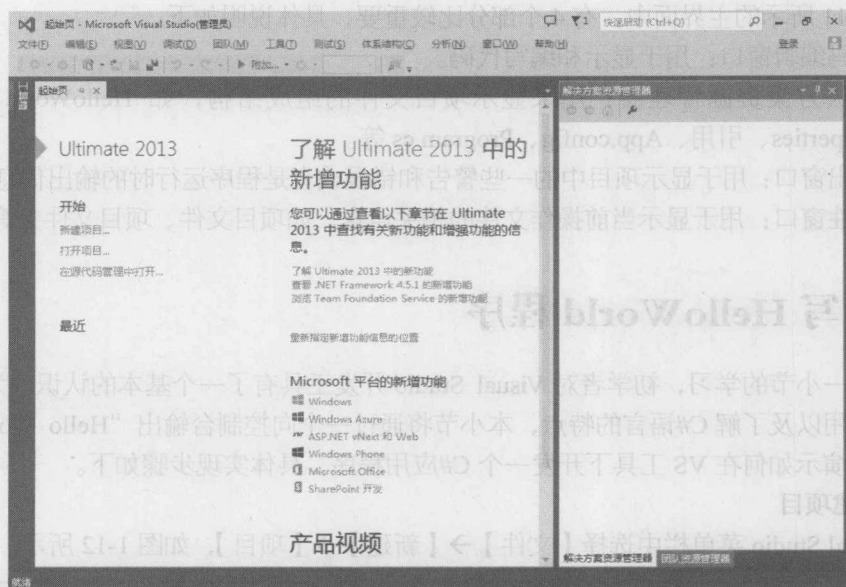


图 1-10 起始页

如图 1-10 所示的起始页中会显示 Visual Studio 的基本信息以及 Visual Studio 的新增功能等内容。

1.2.4 Visual Studio 主界面

使用 Visual Studio 工具进行程序开发，主要是在 Visual Studio 的主界面中进行的。主界面由标题栏、菜单栏、工具栏、代码编辑窗口、解决方案资源管理器、输出窗口、属性窗口等组成，具体如图 1-11 所示。

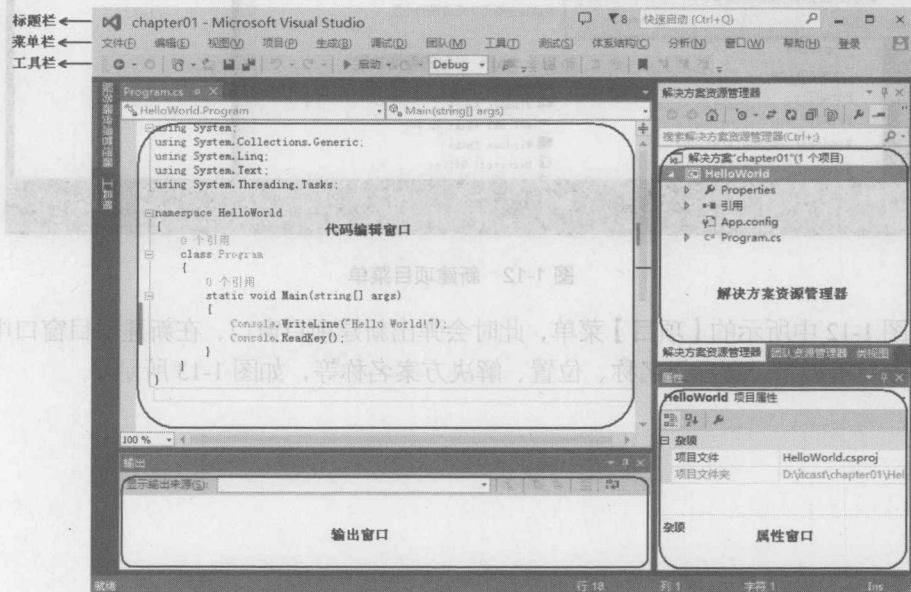


图 1-11 主界面

如图 1-11 所示的主界面中，有 4 个部分比较重要，具体说明如下。

- 代码编辑窗口：用于显示和编写代码。
- 解决方案资源管理器：用来显示项目文件的组成结构，如 HelloWorld 项目中有 Properties、引用、App.config、Program.cs 等。
- 输出窗口：用于显示项目中的一些警告和错误或者是程序运行时的输出信息、异常等。
- 属性窗口：用于显示当前操作文件的相关信息，如项目文件、项目文件夹等。

1.3 编写 HelloWorld 程序

通过上一小节的学习，初学者对 Visual Studio 开发工具有了一个基本的认识。为了快速熟悉工具的使用以及了解 C# 语言的特点，本小节将通过一个向控制台输出“Hello World!”的程序为初学者演示如何在 VS 工具下开发一个 C# 应用程序，具体实现步骤如下：

1. 新建项目

在 Visual Studio 菜单栏中选择【文件】→【新建】→【项目】，如图 1-12 所示。

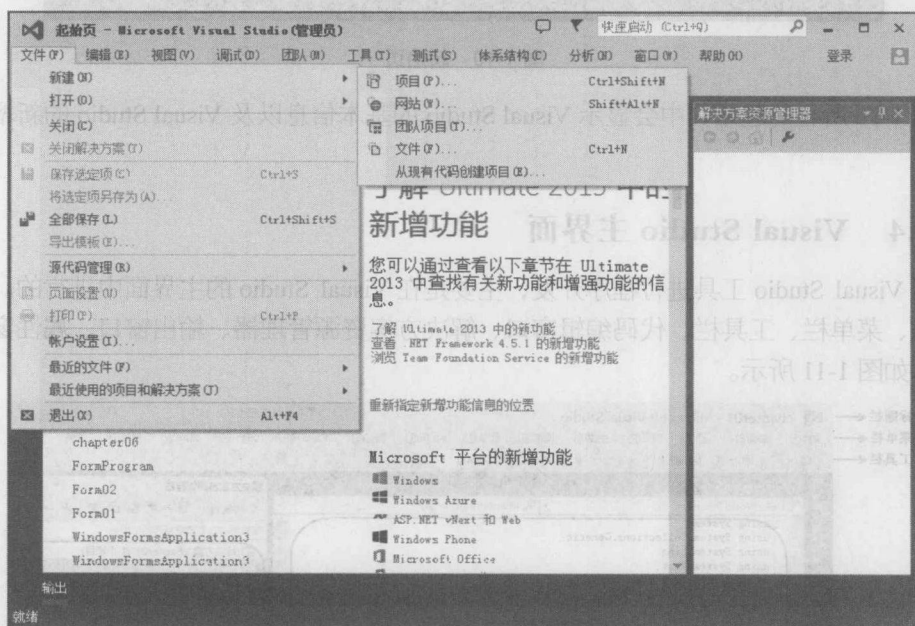


图 1-12 新建项目菜单

点击图 1-12 中所示的【项目】菜单，此时会弹出新建项目窗口，在新建项目窗口中可以选择创建的项目类型，设置项目名称、位置、解决方案名称等，如图 1-13 所示。