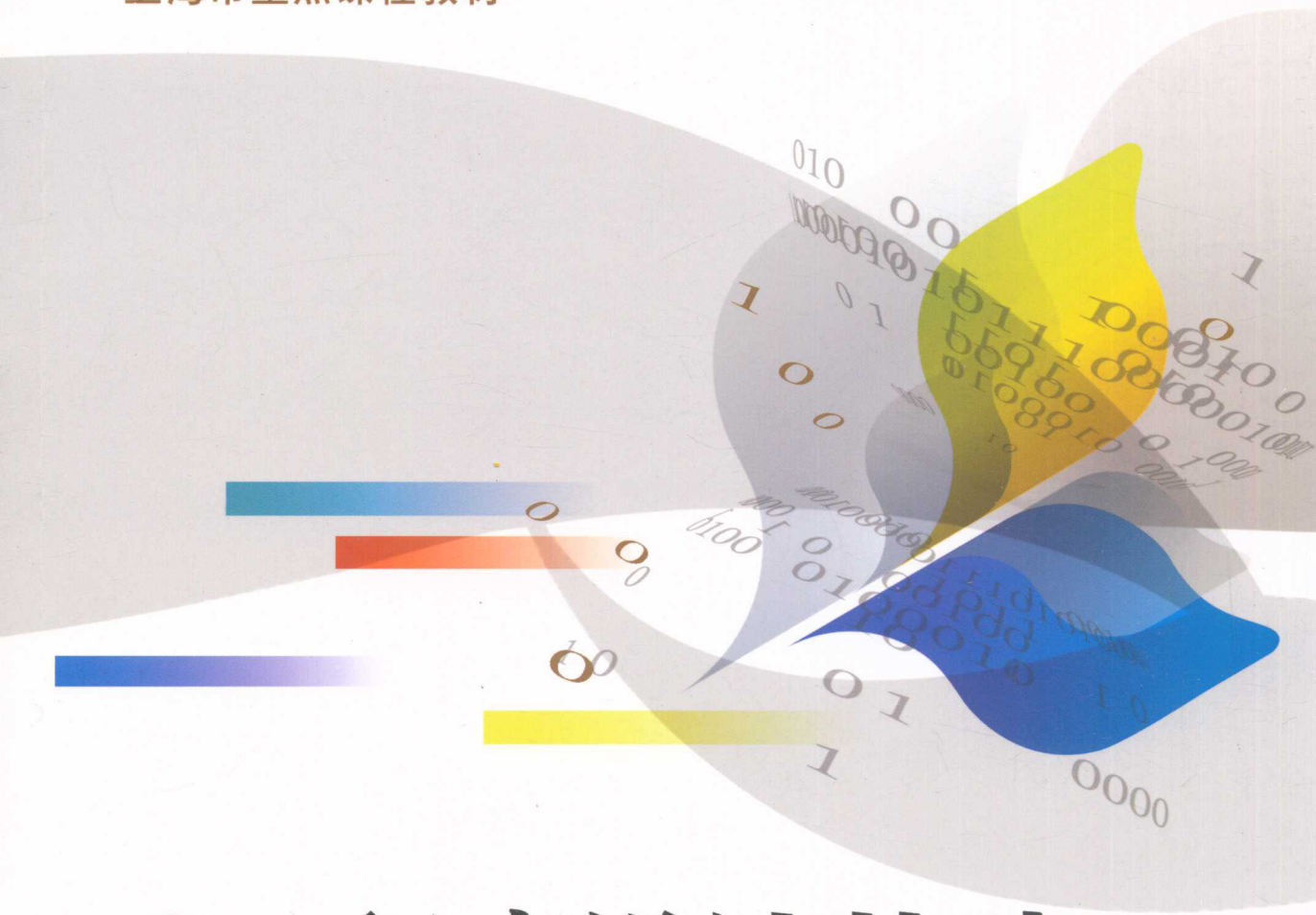
上海市重点课程教材



C#程序设计基础

C# Programming

◎ 张世明　编著

中国工信出版集团
電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

上海市重点课程教材

C#程序设计基础

张世明　编著

電子工業出版社
Publishing House of Electronics Industry
北京・BEIJING

内 容 简 介

本书采用 Microsoft Visual Studio 2010 体系和环境，系统、全面、深入地介绍使用 C#进行 WinForm 应用程序开发应该掌握的各方面技术。本书结合可视化的编程方法和面向对象的编程方法，选取简短、易学的实用例子帮助读者深入理解所学的内容。

本书共 10 章，主要内容包括：C#语言开发环境、C#语言基础、面向对象程序设计、常用标准控件的使用、Windows 应用程序开发、GDI+编程、多线程编程、数据库编程、文件操作等。每章后面都提供了实验和习题，使读者充分掌握每个知识点。为方便教学，本书配有免费电子课件。

本书由浅入深地介绍了编写功能齐备的应用程序所需要的各个组成部分，简洁的语言、完整的代码和详细的分析使读者能够真正体会到 C#的强大功能，同时使 C#的学习更加轻松和高效。

本书可作为普通高等学校“C#程序设计”、“可视化程序设计”等相关课程的教材，也可作为软件开发人员的技术参考书籍。

图书在版编目（CIP）数据

C#程序设计基础 / 张世明编著. —北京：电子工业出版社，2016.2
ISBN 978-7-121-28185-3

Ⅰ. ①C… Ⅱ. ①张… Ⅲ. ①C 语言－程序设计－高等学校－教材 Ⅳ. ①TP312

中国版本图书馆 CIP 数据核字（2016）第 030739 号

策划编辑：冉 哲
责任编辑：冉 哲
印 刷：三河市鑫金马印装有限公司
装 订：三河市鑫金马印装有限公司
出版发行：电子工业出版社
北京市海淀区万寿路 173 信箱 邮编 100036
开 本：787×1092 1/16 印张：19.75 字数：505 千字
版 次：2016 年 2 月第 1 版
印 次：2016 年 2 月第 1 次印刷
印 数：3000 册 定价：39.80 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线：（010）88258888。

前　言

C#是一种安全的、稳定的、简单的、高效的、由 C 和 C++语言衍生出来的面向对象的编程语言。它在继承 C 和 C++语言强大功能的同时，去掉了一些复杂特性。C#语言综合了 Visual Basic 语言的简单的可视化操作和 C++语言的高运行效率，借鉴了 Delphi 语言与 COM 直接集成的特点，以其强大的操作能力、优雅的语法风格、创新的语言特性和便捷的面向组件编程的支持成为.NET 开发的首选语言。

C#还深受其他语言的影响，包括 Java 和 Delphi 语言，其功能的健全毫不逊色于其他面向对象语言，非常适合初学者学习程序设计。

C#博采众家之长，同时克服了它们各自的缺点。Windows 下的开发工具众多，而 C#开发环境是目前面向对象和控件开发性能最好的工具之一。它使用 Windows 图形用户界面的许多先进特性和设计思想，采用可重复利用的、完整的、面向对象的程序设计语言。

本书采用 Microsoft Visual Studio 2010 体系和环境，对使用 C#语言进行编程提供基础性的指导和参考。读者使用本书不需要预先具有任何编程经验，但如果对 C、C++、Java 或 Delphi 等语言有所了解，那么将会对本书的学习非常有益。本书逐步将 C#技术介绍给读者，从入门到提高，同时对 C#的面向对象和可视化组件进行了说明，读者将学会如何为各种应用程序创建切实可行又简洁漂亮的用户界面。

“边学边做”是学习编程语言的最有效方法，希望读者在阅读本书的过程中能够上机实践。每学完一个例子，尝试着改变一些，或者添加一些东西，或者修改一些代码，将有助于读者体验进步和成功的乐趣。本书配有电子课件，可免费提供给教师进行多媒体教学和读者自学时使用。

凡章节、习题的标题中有“*”标记的，表明有一定难度，读者可选择学习。

本书由张世明编著，曹晓夏和翁雯参与了部分章节的编写，在撰写过程中得到了同行专业人士的指点和帮助，在此表示感谢。

如果读者在本书中发现了错误或有疑问，可以给作者发电子邮件（vcline@sohu.com），或者联系本书的责任编辑冉哲（ran@phei.com.cn）。

作者

目　录

第 1 章　C#语言开发环境

【教学提示】Visual Studio.NET 是一个集成开发环境，从程序设计、代码编译和调试，到最后形成发布程序的全部工作都可以在这个集成环境中完成。为了帮助用户快速、高效地开发应用程序，Visual Studio.NET 提供了许多功能强大的开发工具和丰富的菜单命令。本章将对 Visual Studio.NET 中的 C#集成环境的各个组成部分做概要的描述。

【教学要求】在学习 C#语言之前，不仅要清楚 Visual Studio.NET 的总体构成，了解其具有的工具，还要知道其集成环境。本章主要介绍 Visual Studio.NET 集成环境的一些基础知识。通过本章学习，读者应该了解 C#语言的特点及其发展趋势、C#语言简介、C#的应用领域；了解 Visual Studio.NET 具有的工具，熟悉其集成环境；掌握安装 Visual Studio 2010 的硬件要求和安装步骤；熟悉 Visual Studio 2010 开发环境，包括编辑器和 Windows 窗体设计器、解决方案资源管理器和项目设计器、编译器、调试器和错误列表窗口、工具箱和属性窗口等；学会使用 Visual Studio 2010 帮助系统；掌握 C#程序设计过程。

1.1　C#语言概述

随着微软.NET 平台的逐步推广，C#作为一种基于该平台的面向对象编程语言，自面世以来便以其易学易用、功能强大的特点得到了广泛的应用，赢得了越来越多的程序开发人员的喜爱；它与 C++、Java 语言类似，但又有所改进，是一种简单、现代、高效、面向对象、类型安全的新型编程语言，开发人员可以通过它编写在.NET Framework 上运行的各种安全可靠的应用程序，不但可以开发数据库管理系统，而且也可以开发多媒体应用程序和网络应用程序。而 Visual Studio 开发平台则凭借其强大的可视化用户界面设计，使程序员从复杂的界面设计中解脱出来，让编程成为一种享受。本书中涉及的程序都是通过 Visual Studio 2010 开发环境编译的。

1.1.1　C#语言的特点

C#是一种面向对象的编程语言，主要用于开发可以在.NET 平台上运行的应用程序。C#语言是从 C 和 C++语言派生出来的，其语言体系都构建在.NET 框架上，并且能够与.NET 框架完美结合。C#语言具有以下突出的特点。

（1）语法简洁

C#语言主要从 C 和 C++语言继承而来，摒弃了 C 和 C++语言中一些比较复杂而且不常用的语法元素，同时吸收了 Java 语言和 Delphi 语言的优点，它在设计时考虑了多数实际应用的需求。

C#语言取消了指针，不允许直接对内存进行操作，使代码运行在安全的环境中。

字符串定义和操作汲取了 Delphi 语言的 string 数据类型方便性的特点。“::”和“->”这两个域操作符和参照操作符也被“.”操作符所取代。

语法中的冗余是 C++语言中的常见的问题，如 const 和 #define、各种各样的字符类型等。

C#对此进行了简化，只保留了常见的形式，而其他冗余形式从它的语法结构中被清除了出去。

（2）面向对象

C#语言是彻底的面向对象程序设计语言，与C++语言相比，它把面向对象提高到了另一个层次。C#语言具有面向对象程序设计语言所应有的一切特性——封装、继承和多态。

C#语言只允许单继承，即一个类不会有多个基类，从而避免了类型定义的混乱。C#语言中没有了全局函数，没有了全局变量，也没有了全局常数，一切都必须封装在一个类之中。这样，代码将具有更好的可读性，并且减少了发生命名冲突的可能。

（3）类型安全

在C++语言中拥有一个指针，就能自由地把它强制转换成为任何类型，只要内存支持的操作，它就可以做。这并不是企业级编程语言的类型安全。

C#语言实施最严格的类型安全，以保护自己及垃圾收集器（Garbage Collector）。C#语言中变量的规则如下。

不能使用没有初始化的变量。对于对象的成员变量，编译器负责清零；而局部变量，则由用户程序负责清零。当用户程序使用一个没有初始化的变量时，编译器会提示怎么做。优点是能够避免由于使用不经初始化的变量计算结果而导致的错误，而用户还不知道这些奇怪的结果是如何产生的。

C#语言取消了不安全的类型转换，不能把一个整型强制转换成一个引用类型（如对象），而当向下转换时，C#验证这种转换是正确的（也就是说，派生类真的是从向下转换的那个类派生出来的）。

边界检查是C#语言的一部分，再也不会出现这种情况：数组实际只定义了 n–1 个元素，却超额使用了 n 个元素。

算术运算有可能溢出终值数据类型的范围。C#语言允许在语句级或应用程序级检测这些运算。在允许检测溢出的情况下，当溢出发生时将会抛出一个异常。

在C#语言中，被传递的引用参数是类型安全的。

（4）与Web的紧密结合

.NET中新的应用程序开发模型意味着越来越多的解决方案需要与Web标准相统一，例如超文本置标语言（Hypertext Markup Language，HTML）和XML。由于历史的原因，现存的一些开发工具不能与Web紧密地结合。SOAP的使用使得C#语言克服了这一缺陷，大规模深层次的分布式开发从此成为可能。

由于有了Web服务框架的帮助，对程序员来说，网络服务看起来就像是C#的本地对象一样。程序员能够利用他们已有的面向对象的知识与技巧开发Web服务。仅需要使用简单的C#语言结构，C#控件将能够方便地为Web服务，并允许它们通过Internet被运行在任何操作系统上的任何语言所调用。举个例子，XML已经成为网络中数据结构传递的标准，为了提高效率，C#语言允许直接将XML数据映射成为结构。这样就可以有效地处理各种数据。

（5）版本处理

在过去的几年中，几乎所有的程序员都至少有一次不得不涉及众所周知的“DLL异版本调用异常”。该问题起因于多个应用程序都安装了相同DLL名字的不同版本。有时，老版本的应用程序可以很好地和新版本的DLL一起工作，但是更多的时候它们会中断运行。现在的版本问题真是令人头痛。

C#语言的NGWS runtime将对所写的应用程序提供版本支持。C#语言可以很好地支持版本

控制。尽管 C#语言不能确保正确的版本控制，但是它可以为程序员保证版本控制成为可能。有这种支持，一个开发人员就可以确保当他的类库升级时，仍保留着对已存在的客户应用程序的二进制兼容。

（6）兼容性和灵活性

C#语言并没有存在于一个封闭的世界中。它允许使用最先进的 NGWS 的通用语言规定（Common Language Specification，CLS）访问不同的 API。CLS 规定了一个标准，用于符合这种标准的语言内部之间的操作。为了加强 CLS 的编译，C#编译器检测所有的公共出口编译，并在通不过时列出错误。

当然，C#语言还能够访问一些旧的 COM 对象。NGWS 运行时提供对 COM 透明的访问。OLE 自动化是一种特殊的机制，使用 C++语言创建 OLE 自动化项目的人已经喜欢上各种各样的自动化数据类型，C#语言支持它们，而没有烦琐的细节。最后，C#语言允许用 C 原型的 API 进行内部操作。可以从应用程序访问任何 DLL 中的入口点（有 C 原型）。用于访问原始 API 的功能称为平台调用服务（Platform Invocation Services，缩写 PInvoke）。

尽管 C#代码的默认状态是类型安全的，但是可以声明一些类或者仅声明类的方法是非安全类型的。这样的声明允许使用指针、结构，静态地分配数组。安全码和非安全码都运行在同一个管理空间内，这样意味着当从安全码调用非安全码时不会陷入列集（Marshalling）。

1.1.2 C#语言与.NET 的关系

.NET Framework 是微软公司近年来主推的应用程序开发框架，该框架提供跨平台和跨语言的特性，C#是其主要的开发语言。使用.NET 框架，配合微软公司推出的 Visual Studio 集成开发环境，开发人员可以比以往更轻松地创建出功能强大的应用程序。

.NET Framework 的一个重要的控件就是公共语言运行时（Common Language Runtime，CLR），它是.NET Framework 的基础。可以将公共语言运行时看作一个在程序执行时管理代码的底层环境，它提供内存管理、线程管理和远程处理等核心服务，还强制实施严格的类型安全检查以及可以提高应用程序安全性和可靠性的其他代码正确性控制机制。事实上，代码管理的概念是公共语言运行时的基本原则。以公共语言运行时为目标的代码称为托管代码，而不以公共语言运行时为目标的代码称为非托管代码。

.NET Framework 的另一个主要控件就是框架类库，它支持多种应用程序的开发，这些应用程序包括控制台应用程序、Windows 应用程序、ASP.NET 应用程序、Windows 服务、XML Web 服务、Web 窗体。

C#语言是专门为与.NET Framework 一起使用而设计的。C#就其本身而言只是一种语言，尽管它用于生成面向.NET 环境的代码，但它本身不是.NET 的一部分。在安装 Microsoft Visual Studio 2010 的同时，.NET Framework 也被安装到本地计算机中。在.NET 中，C#语言的使用比例远远大于其他编程语言。

1.2 安装 Microsoft Visual Studio

下面将详细介绍如何安装 Microsoft Visual Studio 2010，其步骤如下。

（1）将 Microsoft Visual Studio 2010 安装盘放到光驱中，光盘自动运行后会进入安装程序

界面；如果光盘不能自动运行，可以双击 setup.exe 可执行文件，应用程序会自动跳转到如图 1-1 所示的Microsoft Visual Studio 2010安装页面。该页面上有两个安装选项，即安装Microsoft Visual Studio 2010 和检查 Service Release，一般使用第一个安装选项。

图 1-1　Microsoft Visual Studio 2010 安装页面

（2）单击第一个安装选项“安装 Microsoft Visual Studio 2010”，进入如图 1-2 所示的 Microsoft Visual Studio 2010 安装向导页面。

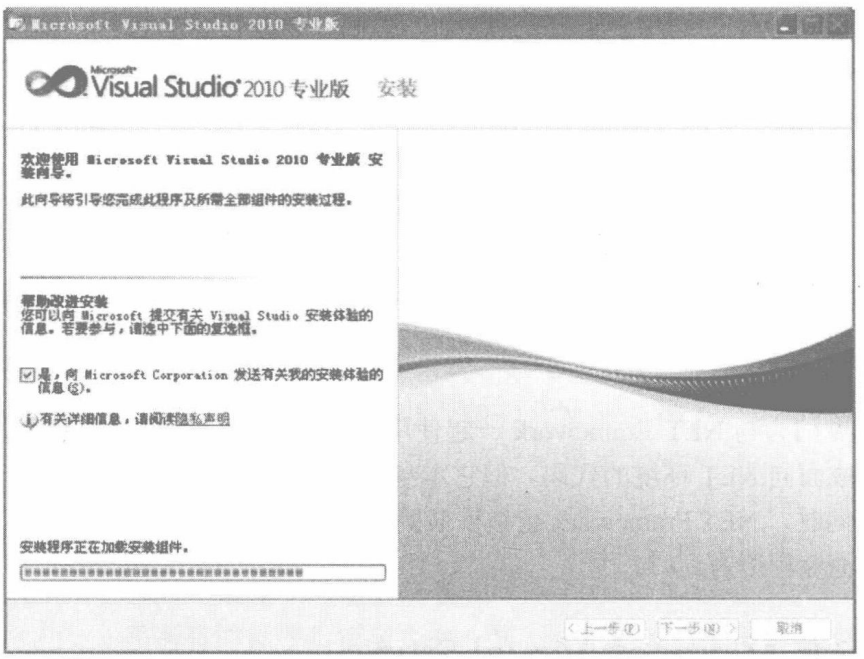


图 1-2　Microsoft Visual Studio 2010 安装向导页面

（3）单击“下一步”按钮，进入如图 1-3 所示的安装程序之起始页。在页面左侧显示的是关于 Microsoft Visual Studio 2010 安装程序所需组件的信息，右侧显示用户许可协议。

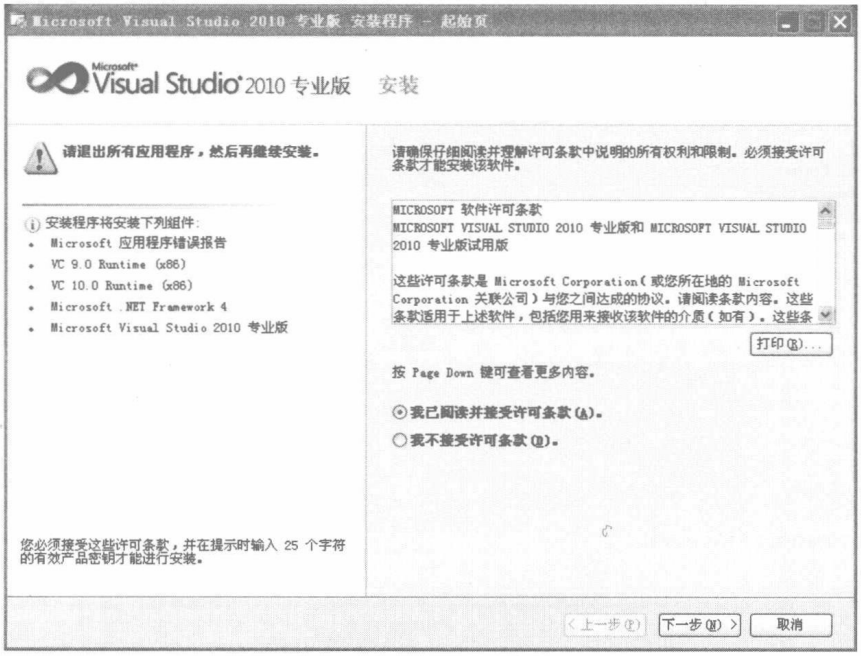


图 1-3　安装程序之起始页

（4）选中“我已阅读并接受许可条款”单选按钮，单击“下一步”按钮，进入安装程序之选项页，用户可以从中选择要安装的功能和产品安装路径（产品安装默认路径为“C:\Program Files\Microsoft Visual Studio 10.0\”），也可以根据需要设置不同的安装路径，如图 1-4 所示。

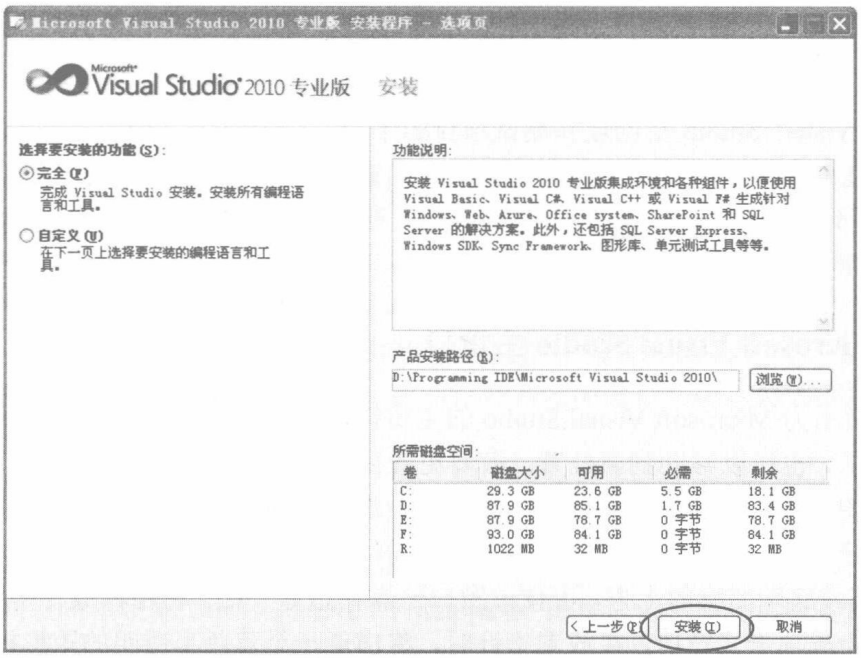


图 1-4　安装程序之选项页

说明：在“选择要安装的功能”栏中，用户可以选择“完全”和“自定义”两种安装方式。如果选择“完全”，安装程序将安装系统的所有功能；如果选择“自定义”，用户可以选择希望安装的项目，增加了安装程序的灵活性。通常，推荐用户选择“完全”。

（5）单击“安装”按钮，进入如图 1-5 所示的安装程序之安装页，显示正在安装控件。

图 1-5　安装程序之安装页

（6）安装完毕后，单击“下一步”按钮，进入安装程序之完成页，单击“完成”按钮，至此完成 Microsoft Visual Studio 2010 开发环境的安装。

1.3　Microsoft Visual Studio 集成开发环境

Microsoft Visual Studio 是创建和测试项目的平台，Microsoft Visual Studio 集成开发环境（IDE）由各种工具组成，包括允许直观创建窗体的窗体设计器、用于输入和修改代码的智能编辑器、把 C#语句翻译成中间机器代码的编译器、帮助查找和修正程序错误的调试器、用来查看对象的属性和方法的对象浏览器及帮助工具等。

1.3.1　Microsoft Visual Studio 主窗口

如图 1-6 所示为 Microsoft Visual Studio 的主窗口，其中的所有窗口都可以变为可停靠或浮动、隐藏或可见，也可以移动到新位置。若要更改窗口的行为，可以单击该窗口标题栏中的向下箭头或图钉图标，然后从中进行选择。若要将停靠窗口移动到新位置，可以拖动标题栏，直至出现一个滴管图标，按住鼠标左键的同时，将指针移至新位置该图标的上方。将指针放在左侧、右侧、顶端或底端图标的上方，可使该窗口停靠在指定一侧。将指针放在中间图标的上方，可使该窗口成为选项卡式窗口。在放置指针时，将出现一个蓝色半透明的矩形框，它指示该窗口将停靠在新位置的什么地方。

Microsoft Visual Studio 主窗口主要通过菜单栏和工具栏进行操作。如果所使用的版本是中文版的，则可以直接通过中文菜单项进行操作；对于工具栏，只要将指针移至工具栏中的某选项上，就会出现该工具栏选项的中文提示（也称为 Hint），所以这里不再对菜单栏和工具栏功能一一做介绍。

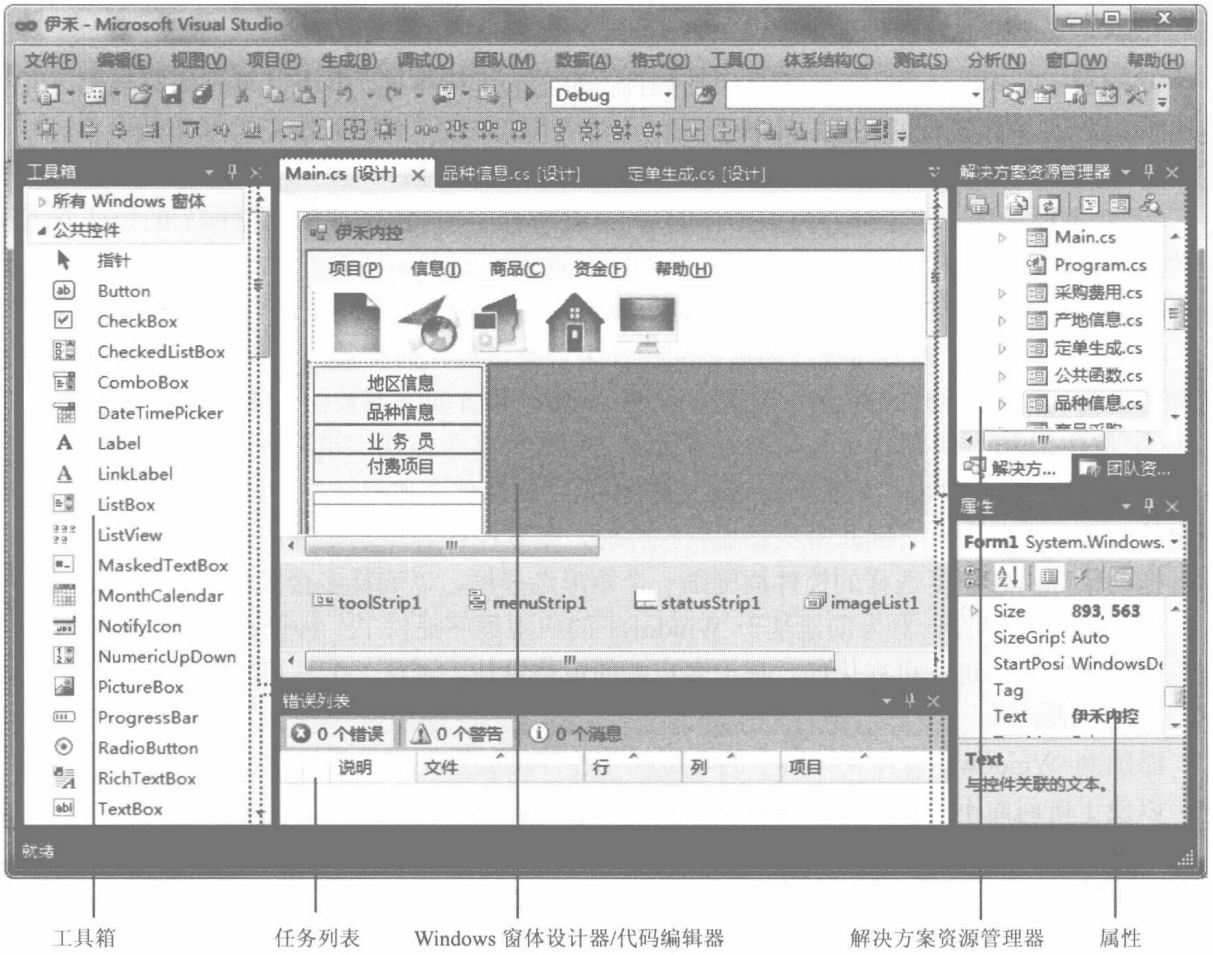


图 1-6 Microsoft Visual Studio 主窗口

1.3.2 代码编辑器与 Windows 窗体设计器

在 Windows 窗体程序设计过程中，代码编辑器用来编写程序代码，Windows 窗体设计器用于控件布局及外观的设计。由于代码编辑器和 Windows 窗体设计器合用一个窗口，因此不能同时看得到，需要通过按 F7 键（或 Shift+F7 组合键），或者选择菜单项“视图”→“代码”（或“设计器”），在代码编辑器和 Windows 窗体设计器之间进行切换。为简便起见，本书中将这两个窗口统称为“编辑器/设计器窗口”。

1. 代码编辑器

代码编辑器是用于编写源代码的字处理程序。就像 Word 对句子、段落和语法提供广泛支持一样，C#代码编辑器也为 C#语法和.NET Framework 提供广泛支持。这些支持可以分为 5 个主要的类别。

① IntelliSense：在将.NET Framework 类和方法输入编辑器中时，不断对其基本文档进行更新，同时还具有自动代码生成功能。

② 重构：随着基本代码在开发项目过程中的演变，智能重构基本代码。

③ 代码段：可以浏览的库，其中包含了频繁重复的代码模式。

④ 波浪下画线：当输入内容时，对拼写错误的单词、错误的语法以及警告情况的可见通知。

⑤ 可读性帮助：大纲显示和着色。

2. Windows 窗体设计器

Windows 窗体设计器用于为窗体添加控件，对它们进行排列，并为其事件编写代码。可以使用 Windows 窗体设计器完成以下操作。

① 向某个窗体添加控件、数据控件或基于 Windows 的控件。

② 双击设计器中的窗体并为该窗体编写 Load 事件中的代码，或双击窗体上的控件并为该控件的默认事件编写代码。

③ 通过选择某个控件并输入一个名称来编辑该控件的 Text 属性。

④ 通过使用鼠标或方向键移动选定的控件来调整该控件的位置。同样，使用 Ctrl 键和方向键可更精确地调整控件的位置。最后，使用 Shift 键和方向键调整控件的大小。

⑤ 按住 Shift 键并单击或按住 Ctrl 键并单击可选择多个控件。当对齐或操控控件大小时，如果按住 Shift 键并单击，则第一个选定的控件是主导控件。如果按住 Ctrl 键并单击，则最后选定的控件是主导控件，因此每次添加新控件时，主导控件会随之更改。或者，也可以单击窗体，拖动鼠标，在想要选择的控件周围画一个矩形选择框，以选择多个控件。

Windows 窗体设计器为创建基于 Windows 的应用程序提供了一种快速的开发解决方案。在该设计器中可以进行可视化的、基于客户端的窗体设计。可从“工具箱”中将控件拖动或绘制到其图面上。

添加到 Windows 窗体设计器中的非可视化控件将放置在位于设计图面下方的“控件栏”中，以便于访问而不干扰可视化设计空间。

此外，在 Windows 窗体设计器中右击会显示一个快捷菜单，可以使用其中的命令切换到代码编辑器，将控件锁定到窗体上，或查看窗体的属性。

1.3.3 解决方案资源管理器和项目设计器

1. 解决方案资源管理器

解决方案资源管理器以分层树视图的方式显示项目中的所有文件。如果使用“项目”菜单项将新文件添加到项目中，将看到这些文件显示在解决方案资源管理器中。除文件外，解决方案资源管理器还显示项目设置，以及对应用程序所需的外部库的引用。

2. 项目设计器

右击解决方案资源管理器中的“Properties”节点，然后在快捷菜单中选择“打开”命令，将访问“项目设计器”，如图 1-7 所示。使用其中的属性页可以修改生成选项、安全要求、部署详细信息以及许多其他项目属性。

若要访问单个文件的属性，应使用属性窗口。

在项目设计器中可以集中管理项目的属性、设置和资源。与其他设计器（如窗体设计器或类设计器）一样，项目设计器在 Microsoft Visual Studio IDE 中作为单一窗口出现。

1.3.4 编译器、调试器和错误列表窗口

1. 编译器

C# 编译器没有窗口，因为它不是交互式工具，但可以在项目设计器中设置编译器选项。

选择菜单项“生成”→“生成”，IDE 将调用 C#编译器。如果生成成功，则在状态窗格中将显示“生成成功”消息。如果存在生成错误，将在编辑器/设计器窗口的下方出现“错误列表”窗口。双击其中某个错误可以转到源代码中相应的问题行。按 F1 键可以查看针对突出显示的错误的帮助文档。

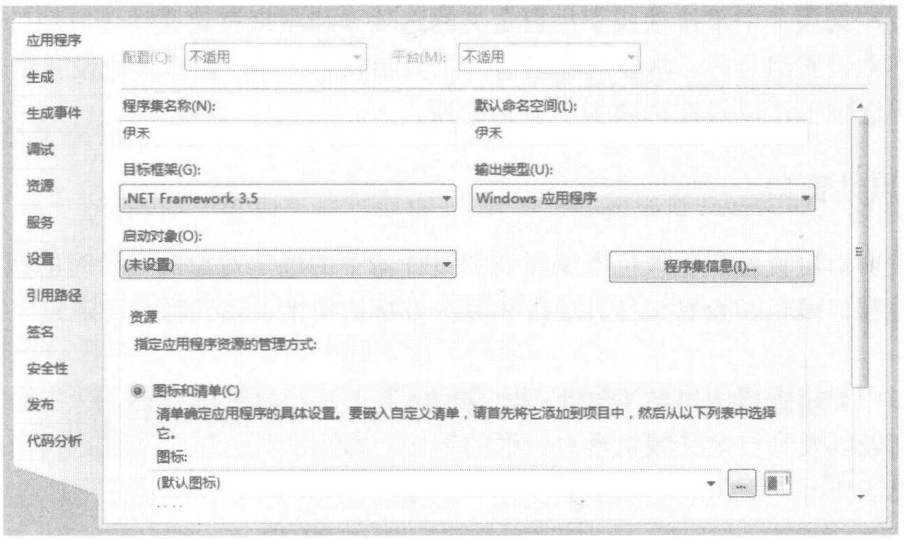


图 1-7　项目设计器

2．调试器

调试器具有多个不同的窗口，这些窗口随着应用程序的运行而显示不同变量的值和类型信息。在调试器中调试时，可以使用代码编辑器指定在某一行暂停执行，以及每次一行单步执行代码。

3．错误列表窗口

“错误列表”有助于用户加快应用程序开发的过程。在“错误列表”窗口中，可以进行以下操作。

① 显示在编辑和编译代码时产生的错误、警告和消息。

② 查找 IntelliSense 所标出的语法错误。

③ 查找部署错误、某些静态分析错误以及在应用“企业级模板”策略时检测到的错误。

④ 双击任意错误信息项打开出现问题的文件，然后转移到错误位置。

⑤ 筛选显示哪些项，以及为每项显示哪些列的信息。

要显示“错误列表”窗口，选择菜单项“视图”→“错误列表”。

1.3.5　工具箱

“工具箱”是一个浮动的树控件，它与 Windows 资源管理器的工作方式非常类似，但没有网格或连接线。可以同时展开工具箱的多个页（称为“选项卡”），整个目录树在工具箱内部滚动。要展开工具箱的任何选项卡，单击它名称旁边的加号(+)；要折叠一个已展开的选项卡，单击它名称旁边的减号(–)。

工具箱中显示可以添加到项目中的项的图标。每次返回编辑器或设计器时，工具箱都会自