



高等院校信息类专业校企衔接创新实践系列教材

.NET 项目开发教程

□ 主 编 余秋明
□ 副主编 陈璠 朱勇 张小平



中南大学出版社
www.csupress.com.cn

非
外
借

.NET 项目开发教程

主 编 余秋明

副主编 陈 璞 朱 勇 张小平



中南大学出版社

www.csupress.com.cn

· 长 沙 ·

图书在版编目 (C I P) 数据

.NET 项目开发教程 / 余秋明主编. --长沙: 中南
大学出版社, 2017.7

ISBN 978 -7 -5487 -2881 -8

I. ①N… II. ①余… III. ①网页制作工具—程序设
计—教材 IV. ①TP393.092.2

中国版本图书馆 CIP 数据核字(2017)第 175844 号

.NET 项目开发教程

.NET XIANGMU KAIFA JIAOCHENG

余秋明 主编

☐责任编辑 胡小锋

☐责任印制 易建国

☐出版发行 中南大学出版社

社址: 长沙市麓山南路

邮编: 410083

发行科电话: 0731 - 88876770

传真: 0731 - 88710482

☐印 装 长沙市宏发印刷有限公司

☐开 本 787 × 1092 1/16 ☐印张 20.75 ☐字数 528 千字

☐版 次 2017 年 7 月第 1 版 ☐2017 年 7 月第 1 次印刷

☐书 号 ISBN 978 -7 -5487 -2881 -8

☐定 价 59.00 元

图书出现印装问题, 请与经销商调换

前 言

本教材是在学生学习过 C#程序设计课程理论知识之后侧重于真实的软件项目开发的教材,可用于专业学生软件工程实践类课程,也可作为软件开发人员基于 .NET 项目实战开发参考。

本教材先是对 C#语法总结及知识点精讲,在学生具备一定的 C#语法知识的基础上再重点描述一个完整的真实软件项目开发过程,即采用 C#语言基于 .NET 开发网上在线商城软件项目。该项目分阶段描述为项目准备阶段、项目启动阶段、系统分析与设计阶段、系统编码阶段、测试阶段、系统收尾阶段。在项目准备阶段描述了软件开发过程、软件开发流程模型、面向对象程序设计、团队分工、项目环境准备以及如何进行源代码管理;在项目启动阶段如何制订项目目标和项目总体计划;在系统分析与设计阶段如何进行需求调研、需求分析、软件原型设计、系统框架设计、功能用例设计、关键类设计、数据模型设计等;系统编码阶段如何实现编码存储过程,编码如何实现系统框架、各核心功能模块并进行单元测试等;在系统测试阶段如何选用测试工具,如何按照测试计划进行系统测试并撰写测试报告;最后要进行项目收尾,一般包括制作安装程序(包括制作、配置和测试)、系统验收等,验收完成后就进入系统运维阶段。

通过使用该教材,并在项目老师的指导下,学生能身临其境地、全阶段式参与项目并实战式地进行软件项目开发训练。相信学生能够快速提升项目开发能力,积累一定的软件开发经验,适应软件项目开发过程中的角色定位,从而为日后从事软件开发工作奠定基础。

本教材具有以下特点:

1. 注重语法基础与提高:本教材前面章节从学生已经学过 C#程序设计课程的角度出发,对软件开发技术中使用到的语法知识进行总结和归纳,并有侧重点地进行知识点精讲,由浅入深,逐步提高编程的基础知识;

2. 注重项目实战描述:本教材重点突出描述完整的真实的软件项目开发过程,让学生不仅仅了解基础知识、编程知识,而且重点了解软件是怎么开发出来的,软件项目是怎样进行设计、实现、管理的;

3. 注重学生亲自参与:本教材编写的真实的软件项目开发过程,不仅仅让学生了解其开发过程,重点还在于设计的项目学生能够在指导老师的指导下亲自动手,身临其境地全程参与进来,每个软件开发阶段学生都可亦步亦趋地学习,亲身体会软件开发过程,进行软件项

目实战。

本教材由余秋明、陈瓏编写第一篇章，中软国际(厦门)公司教学总监朱勇、教学部讲师张小平编写第二篇章，最后由余秋明负责全书统稿。

由于编者水平有限,书中难免存在疏漏和不足之处,敬请专家与读者批评指正。

编者

2017 年 9 月

目 录

第一篇 技术储备篇

第 1 章 .NET 平台概述	(3)
1.1 .NET 概述	(3)
1.2 C#语言概述	(5)
1.3 熟悉 Visual Studio 开发环境	(7)
第 2 章 C#基础知识	(10)
2.1 变量	(10)
2.2 表达式和运算符	(15)
2.3 程序流程控制	(17)
2.4 类及对象	(20)
2.5 类成员	(23)
第 3 章 ADO.NET	(32)
3.1 ADO.NET 概述	(32)
3.2 ADO.NET 访问数据库一般步骤	(35)
3.3 DataSet 数据查询与更新	(50)

第二篇 项目实战篇

第 4 章 准备阶段	(63)
4.1 软件开发过程	(63)
4.2 软件开发流程模型	(65)

4.3	面向对象分析设计	(70)
4.4	团队分工	(74)
4.5	项目环境	(78)
4.6	源代码管理	(79)
第5章	项目启动阶段	(96)
5.1	项目案例介绍	(96)
5.2	项目范围	(96)
5.3	项目总体计划	(97)
第6章	系统分析与设计阶段	(103)
6.1	需求调研	(103)
6.2	需求分析	(105)
6.3	原型设计	(113)
6.4	系统框架设计	(121)
6.5	功能用例设计	(122)
6.6	关键类设计	(132)
6.7	数据模型设计	(133)
第7章	系统编码阶段	(141)
7.1	存储过程实现	(141)
7.2	系统框架实现	(143)
7.3	单元测试	(167)
7.4	登录页面	(173)
7.5	网站主页	(188)
7.6	核心功能模块实现	(208)
7.7	报表实现	(292)
第8章	测试阶段	(300)
8.1	测试工具	(300)
8.2	测试计划	(302)
8.3	系统测试	(303)
8.4	测试报告	(306)

第 9 章 收尾阶段	(308)
9.1 安装制作	(308)
9.2 系统验收	(317)
9.3 系统运维	(321)
参考文献	(323)

第一篇 技术储备篇

第1章 .NET 平台概述

1.1 .NET 概述

.NET 平台指的是.NET 框架(.NET Framework)。。.NET 框架(.NET Framework) 是由微软开发的, 一个致力于敏捷软件开发(Agile Software Development)、快速应用开发(Rapidapplication Development)、平台无关性和网络透明化的软件开发平台。它不仅是一个软件开发平台, 同时也是.NET 开发出来的程序必备运行环境。.NET Framework 支持生成和运行下一代应用程序和 XML Web Services 的内部 Windows 组件。Microsoft 正在五个方面创建.NET 平台, 即工具、服务器、XML Web 服务、客户端和.NET 体验。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, 提供从开发、管理、使用到体验 XML Web 服务的方方面面。

1.1.1 .NET Framework

.NET Framework 的目的就是要让建立 Web Services 以及因特网应用程序的工作变得简单。.NET Framework 的体系结构包括5大部分, 它们是:

- (1) 程序设计语言及公共语言规范(CLS);
- (2) 应用程序平台(ASP.NET 及 Windows 应用程序等);
- (3) ADO.NET 及类库;
- (4) 公共语言运行库(CLR);
- (5) 程序开发环境(Visual Studio.NET)。

其结构如图 1-1 所示。

.NET Framework 旨在实现下列目标:

- (1) 提供一个一致的面向对象的编程环境, 无论对象代码是在本地存储和执行的, 还是在本地执行但在 Internet 上分布的, 或者是在远程执行的。
- (2) 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。
- (3) 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。
- (4) 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。



图 1-1 .NET Framework 体系结构

1.1.2 公共语言运行库 (CLR)

公共语言运行库是 .NET Framework 的基础，它为多种语言(例如 C#、VB、VC++ 等)提供了一种统一的运行环境。可以将公共语言运行库看做是一个在执行程序时进行代码管理的“工具”，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。托管代码具有许多优点，如跨语言集成、跨语言异常处理、增强的安全性、调试和分析服务等。公共语言运行库结构如图 1-2 所示。

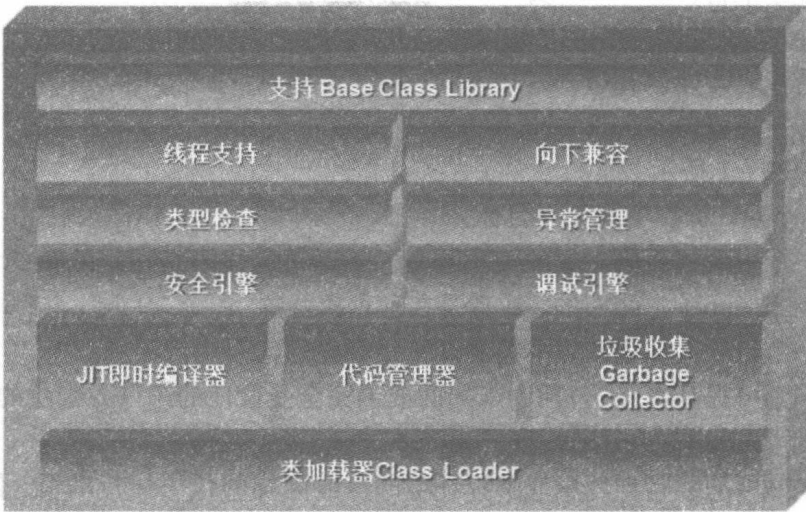


图 1-2 公共语言运行库结构

1.1.3 .NET 程序编译、运行过程

.NET 的编译和运行过程与传统代码类似，但有自己的独特之处。无论是 C#语言，还是 VB.NET 语言，编写的程序代码（称为 .NET 托管代码），都是先被编译成中间语言 IL（Intermediate Language），在运行时，再由即时编译器（JIT，Just - In - Time）编译成本机代码，同时 .NET 代码运行时有一个 CLR 环境来管理程序。NET 代码编译运行过程如图 1 - 3 所示。

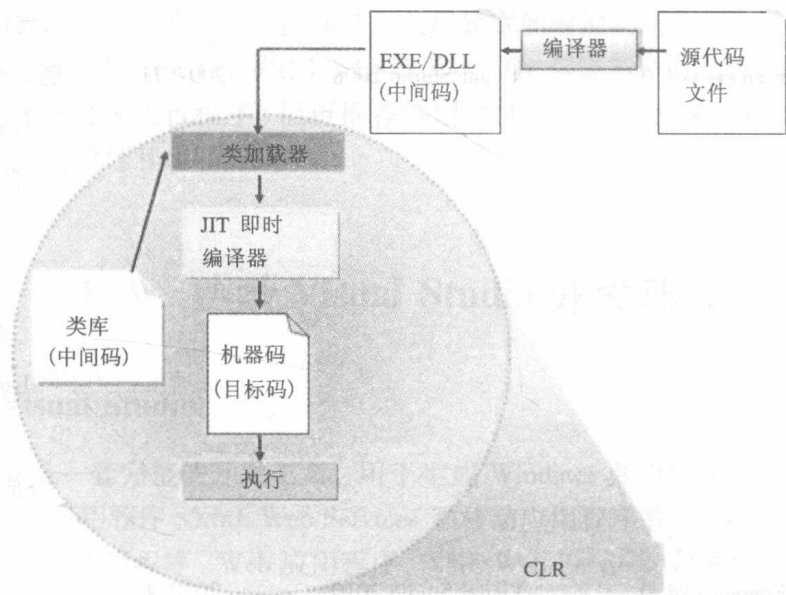


图 1 - 3 .NET 代码编译运行过程

1.2 C#语言概述

1.2.1 C#发展历程

C#是微软公司推出的一种语法简洁、类型安全的面向对象的编程语言，是由 C 和 C++ 衍生出来的。它在继承 C 和 C++ 强大功能的同时去掉了它们的一些复杂特性（例如没有宏以及不允许多重继承）。C#综合了 VB 简单的可视化操作和 C++ 的高运行效率，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#是微软公司在 2000 年 6 月发布的全新编程语言。在其诞生后的十多年里，为了帮助开发人员更好地使用 C#语言来编写应用程序，微软不断地更新 C#语言的版本，每次升级都能带来让用户眼前一新特性的新特性。表 1 - 1 列出了 C#每个版本中所更新的特性和对应的 .NET Framework 版本。

表 1-1 C#各版本的新特性及对应的.NET Framework 版本

C#版本	.NET Framework 版本	Visual Studio 版本	发布日期	特性
C#1.0	.NET Framework1.0	Visual Studio. NET 2002	2002.1	委托事件
C#1.1	.NET Framework1.1	Visual Studio. NET 2003	2003.4	APM
C#2.0	.NET Framework2.0	Visual Studio 2005	2005.11	泛型 匿名方法 迭代器可空类型
C#3.0	.NET Framework3.0	Visual Studio 2008	2007.11	隐式类型的局部变量 对象集合初始化 自动实现属性 匿名类型 扩展方法 查询表达式 Lambda 表达式 表达式树 分部类和分部方法 Linq
C#4.0	.NET Framework4.0	Visual Studio 2010	2010.4	动态绑定 命名和可选参数 泛型的协变和逆变 互操作性
C#5.0	.NET Framework4.5	Visual Studio 2012	2012.8	异步和等待调用

从表 1-1 可以看出，对于 C#的每一个版本，微软都是围绕某个主题进行更新的

1.2.2 C#语言特点

- 用 C#语言进行程序设计的目的是为了与.NET 框架协同工作，它具有以下主要特征。
- (1)简洁的语法。不允许直接操作内存，去掉了指针操作。
 - (2)精心的面向对象设计。C#具有面向对象语言应有的一切特征：封装、继承和多态。
 - (3)与 Web 的紧密结合。C#支持绝大多数的 Web 标准，如 HTML、XML、SOAP。
 - (4)完整的安全性和错误处理。可以消除软件开发中的常见错误(如语法错误)，.NET 提供的垃圾回收器能够帮助开发者有效地管理内存资源。
 - (5)版本处理技术。C#在语言中内置了版本控制功能，提供了接口和接口继承的支持，这些特征可以保证复杂的软件能被方便地开发和升级。
 - (6)灵活性和兼容性。因为 C#遵循.NET 的公共语言规范(CLS)，从而保证能够与其他语言开发的组件兼容。

1.2.3 C#语言编程环境

目前,开发和运行 C#程序有多种选择,例如,用户可以从微软免费获取 .NET 的软件开发工具箱(SDK)或购买功能强大的 Visual Studio .NET 开发环境,各自的特点如下:

SDK 包含编译、运行和测试 C#程序的所有资源,它包含 C#语言编译器、JIT 编译器和相关文档。唯一不含有的是用来输入和编辑 C#程序的文本编辑器。

Visual Studio .NET 是微软的完整开发环境,它包含一个集成开发环境(IDE)和高级 C#编辑器,同时还支持程序调试及许多可提高开发人员效率的附加功能。

Visual Studio .NET 和 SDK 使用相同的 C#编译器、JIT 编译器和 CLR(公共语言运行环境)等程序调试工具来编译和运行程序,但更推荐使用提供了工具包的 Visual Studio .NET,因为它可以让用户更轻松设计和编写 C#程序。

1.3 熟悉 Visual Studio 开发环境

1.3.1 认识 Visual Studio

Visual Studio 是一套完整的开发工具,用于生成 Windows 桌面应用程序、控制台应用程序、ASP.NET Web 应用程序、XML Web Services 和移动应用程序等,它提供了在设计、开发、调试和部署 Windows 应用程序、Web 应用程序、XML Web Services 和传统的客户端应用程序时所需要的工具。目前 Visual Studio 最新版本为 2012,但目前使用比较广泛的有 VS2008, VS2010。Visual Studio2008 开发界面如图 1-4 所示。

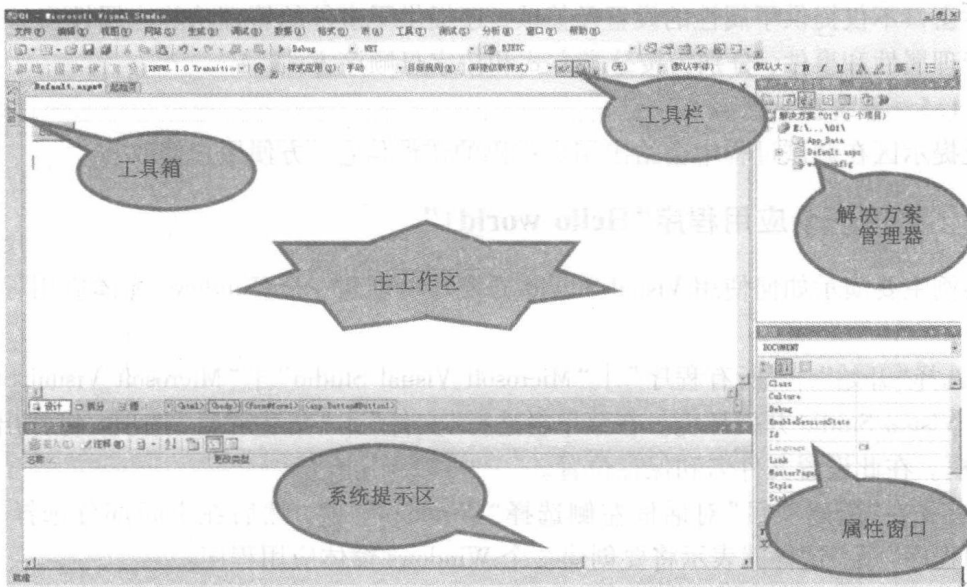


图 1-4 Visual Studio 2008 界面

1.3.1.1 工具栏

为了操作更方便、快捷，菜单项中常用的命令按功能分组分别放入相应的工具栏中。通过工具栏可以迅速地访问常用的菜单命令。常用的工具栏有标准工具栏和调试工具栏。

(1) 标准工具栏包括大多数常用的命令按钮，如新建项目、添加项目、打开文件、保存、全部保存等。

(2) 调试工具栏包括对应用程序进行调试的快捷按钮。

1.3.1.2 工具箱

工具箱提供了进行 .NET 应用程序开发的常用控件。通过工具箱，开发人员可以方便地进行可视化的窗体设计，简化了程序设计的工作量，提高了工作效率。在 Windows 窗体应用程序中，根据控件功能的不同，将工具箱分为 12 个栏目。

单击某个栏目，显示该栏目下的所有控件。当需要某个控件时，可以通过双击所需要的控件直接将控件加载到窗体上，也可以先单击选择需要的控件，再将其拖动到设计窗体上。工具箱中的控件可以通过工具箱右键菜单来控制，如实现控件的排序、删除、显示方式等。

1.3.1.3 主工作区

在 Visual Studio 中主工作区可分为窗体设计器或代码设计器两种模式，通过单击主工作区中的选项卡进行切换。

(1) 窗体设计器：是一个可视化窗口，开发人员可使用鼠标将工具箱中的控件直接拖放到窗体设计窗口中。

(2) 代码设计器：可通过单击“代码设计器”选项卡或双击窗体设计器方式进入代码设计器窗口，开发人员可以在该设计窗口中编写 C# 代码。

1.3.1.4 属性窗口

属性窗口是 Visual Studio 中一个重要的工具，该窗口为 Windows 窗体应用程序的开发提供了简单的属性修改方式。窗体应用程序开发中的各个控件属性都可以由属性窗口设置完成。属性窗口不仅提供了属性的设置及修改，还提供了事件的管理功能。属性窗口采用了两种方式管理属性和事件，分别为按分类方式和按字母顺序方式。

1.3.1.5 系统提示区

系统提示区在调试过程中会给出用户代码的错误信息，方便用户修改。

1.3.2 创建第一个应用程序“Hello world!”

本实例主要演示如何使用 Visual Studio 开发环境创建一个 Windows 窗体应用程序，开发步骤如下：

(1) 选择“开始”|“所有程序”|“Microsoft Visual Studio”|“Microsoft Visual Studio”选项，进入 Visual Studio 开发环境。在菜单栏选择“文件”|“新建项目”选项，将弹出“新建项目”对话框。在此设置文件名和保存位置。

(2) 首先在“新建项目”对话框左侧选择“Windows”项，然后在中间部分选择“Windows 窗体应用程序”项，这样就表示将要创建一个 Windows 窗体应用程序。

(3) 在窗体上放置一个按钮控件，修改其 Text 属性为“单击我”，双击此按钮，进入代码设计器，添加如下代码：

```
private void button1_Click(object sender, EventArgs e)
```

```
    MessageBox.Show("Hello world", "提示信息");
```

具体运行界面如图 1-5 所示。

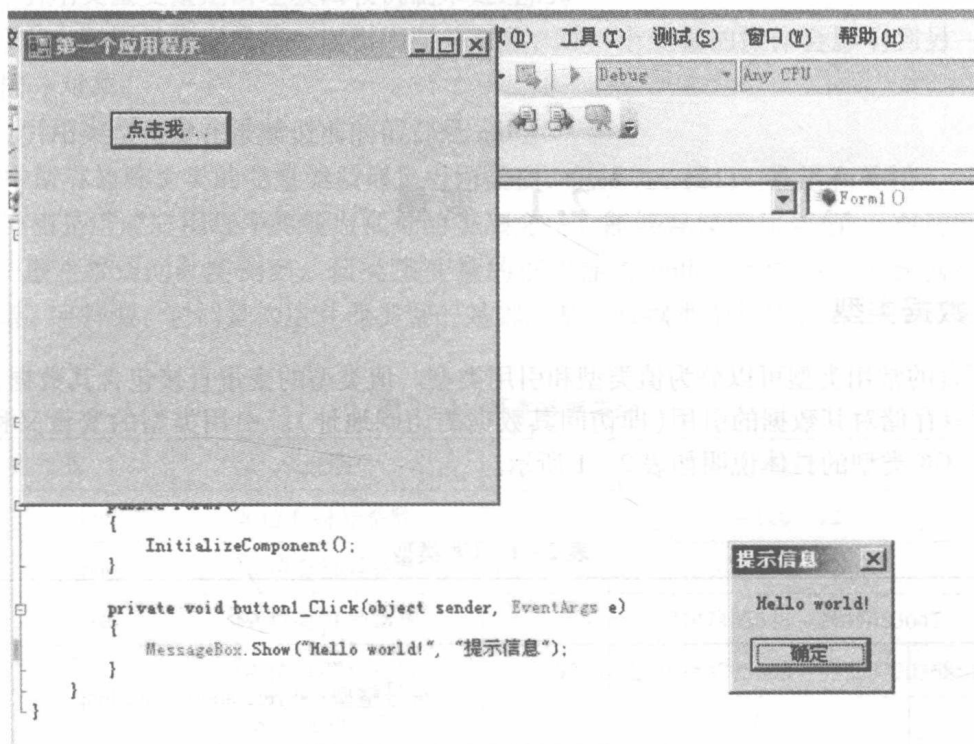


图 1-5 “Hello world”程序运行界面