

• Web编程系列 •

ASP.NET

案例分析与教程

刘宝龙 李 浩 喻 钧 主编

ASP.NET
ANLI FENXI YU JIAOCHENG



科学出版社

Web 编程系列丛书

ASP.NET案例分析与教程

刘宝龙 李 浩 喻 钧 主编

科学出版社

北 京

内 容 简 介

本书从 ASP.NET 初学者的角度出发,对基于 ASP.NET 的 Web 应用开发技术进行详尽介绍。本书共 10 章,分别介绍了 ASP.NET 基础知识、C# 语言基础、ASP.NET 常用控件、ASP.NET 的内置对象、ADO 数据库访问技术、ASP.NET 中数据绑定技术、ASP.NET Web 服务、JavaScript 脚本语言、AJAX 技术基础以及应用案例开发等内容。

全书内容翔实,通俗易懂,适合自学。全书贯穿基于 ASP.NET 的物流信息化平台开发实例,每章内容后配有习题和上机练习,帮助读者深入理解和学习。

本书可作为高等院校计算机应用技术专业、软件工程专业的学生教材,同时也可作为基于 ASP.NET 的 Web 应用开发人员的参考书。

图书在版编目 (CIP) 数据

ASP.NET 案例分析与教程/刘宝龙,李浩,喻钧主编. —北京:科学出版社, 2016.3

(Web 编程系列丛书)

ISBN 978-7-03-047679-1

I. ①A… II. ①刘… ②李… ③喻… III. ①网页制作工具—程序设计—教材 IV. ①TP393.092

中国版本图书馆 CIP 数据核字 (2016) 第 049440 号

责任编辑:杨 岭 李小锐

责任校对:韩雨舟

责任印制:余少力

封面设计:墨创文化

科 学 出 版 社 出 版

北京东黄城根北街 16 号

邮政编码:100717

<http://www.sciencep.com>

成都创新包装印刷厂印刷

科学出版社发行 各地新华书店经销

*

2016 年 3 月第 一 版 开本:16 (787×1092)

2016 年 3 月第一次印刷 印张:19.5

字数:450 千字

定价:49.00 元

(如有印装质量问题,我社负责调换)

前 言

ASP.NET 是在 Microsoft .NET Framework 的基础上构建的, 可提供构建企业级 Web 应用程序所需服务的 Web 平台, 是创建动态交互网页的强有力工具。.NET Framework 是用于构建、开发以及运行 Web 应用程序和 Web Service 的公共环境, 它主要由三部分组成: 编程语言、服务器端和客户端技术、开发环境。ASP.NET 是 .NET Framework 的重要组成部分, 它建立在公共语言运行库上, 可用于在 Web 服务器上生成功能强大的 Web 应用程序, 为 Web 站点创建动态的、交互的 HTML 页面。

作为企业级应用开发的两大主流技术体系之一, .NET 技术近年来发展异常迅速, 越来越受到国内外 IT 企业的认可, 在各行各业都得到了广泛的应用。基于此, 对 .NET 研发人员的需求量也在不断上升, 熟悉 .NET 技术体系的学生就业前景很好。

本书作者长期从事 Web 应用开发和 .NET 技术课程的一线教学工作, 有着深厚的实践开发经验和丰富的教学经验。熟悉 ASP.NET 和 Java EE 两大主流技术体系, 对面向对象技术、设计模式、软件架构等知识理解较为深刻, 能够站在理论的高度来指导实践, 同时, 作者也非常了解学生的认知规律, 进而指导教材编写。

Web 应用开发有着很强的技巧性, 要求学生从整体上把握软件的架构、框架, 合理地使用设计模式, 这样才能设计出稳定性好、扩展性强的软件产品。很多培训公司的课程体系和教材注重实践, 却缺乏理论深度, 培养出来的学生能够应付就业, 却难以取得长远的发展。本教材更注重思想方法的培养, 将面向对象思想、设计模式和软件架构的知识融入各章节教学中, 尽量使学生知其然并知其所以然, 以思想方法指导设计实践。

本书特点:

- (1) 以物流信息化平台为实例, 在各个章节中涉及的案例均以此为基础讲解;
- (2) 第 10 章给出物流信息化平台的完整开发过程及主要的设计文档, 为相关从业人员提供参考;
- (3) 以应用实例为驱动, 使学生在学习的过程中对基于 ASP.NET 的 Web 应用开发有一个全貌的认识;
- (4) 本书以提高学生的实际动手能力为基本出发点, 内容设置贴近实际开发的需要, 且简单易学。

全书所有程序在 Windows XP SP2/Windows 7、IIS 6.0-7.0、.NET Framework 4 下测试通过, 数据库使用 SQL Server 2008, 开发工具采用 Microsoft Visual Studio 2010 旗舰版。

本书第 1、2 章由西安工业大学郭怡老师编写, 第 3、4 章由云南大学李浩老师编写, 第 5、6、10 章由西安工业大学刘宝龙老师编写, 第 7、8、9 章由西安工业大学喻钧老师编写。

尽管在编写本书的过程中尽了最大努力, 但由于编者水平有限, 疏漏及不妥之处在所难免, 恳请读者批评指正。作者联系邮箱: liu.bao.long@hotmail.com, 书中源代码可免费获取。

编 者

2015 年 12 月

目 录

第 1 章 ASP.NET基础	1
1.1 .NET应用开发架构	1
1.1.1 .NET的设计目标	1
1.1.2 Microsoft .NET 框架	1
1.2 ASP.NET概述	3
1.2.1 ASP.NET的脚本语言	4
1.2.2 ASP.NET的工作原理	4
1.3 建立 ASP.NET的运行和开发环境	5
1.3.1 安装和配置 IIS 服务器	5
1.3.2 安装 Visual Studio 开发工具	6
1.3.3 SQL Server 数据库系统的安装	8
1.4 开始编写 ASP.NET程序	9
1.4.1 Web 窗体代码模型	9
1.4.2 ASP.NET网页设计实例	9
1.5 习题与上机练习	11
第 2 章 C# 语言基础	12
2.1 C#概述.....	12
2.1.1 C#的特点.....	12
2.1.2 创建一个简单的 C#程序	12
2.1.3 程序的输入输出	13
2.1.4 C#注释.....	14
2.2 C# 基本语法	15
2.2.1 C# 常量与变量	15
2.2.2 C# 数据类型	15
2.2.3 运算符和表达式	23

2.2.4 程序控制结构	27
2.3 异常处理	33
2.3.1 try-catch 结构	33
2.3.2 try-finally 结构	34
2.3.3 try-catch-finally 结构	34
2.4 类和对象	35
2.4.1 类和对象的创建	35
2.4.2 属性和方法	39
2.4.3 构造函数和析构函数	40
2.4.4 继承和多态	42
2.5 字符串	44
2.5.1 使用字符串	44
2.5.2 创建动态字符串	48
2.6 习题与上机练习	50
第3章 ASP.NET常用控件	53
3.1 ASP.NET页面的生命周期	53
3.2 服务器控件概述	54
3.2.1 理解服务器控件	54
3.2.2 服务器控件的分类	55
3.2.3 服务器控件的共有事件	56
3.2.4 页面指示符	57
3.3 HTML 服务器端控件	57
3.4 标准的 Web 服务器控件	60
3.4.1 文本输入与显示控件	60
3.4.2 控制权转移控件	62
3.4.3 选择控件	65
3.4.4 容器控件	71
3.4.5 表格控件	73
3.4.6 广告栏控件和日历控件	74
3.5 验证控件	76
3.5.1 必须输入验证控件	77
3.5.2 比较验证控件	78

3.5.3 范围验证控件	80
3.5.4 正则表达式验证控件	81
3.5.5 验证总结控件	84
3.5.6 自定义验证控件	85
3.6 用户控件	87
3.6.1 用户控件概述	87
3.6.2 创建用户控件	87
3.6.3 用户控件的使用	89
3.6.4 用户控件实现注册界面	90
3.7 服务器端控件的动态数据绑定	91
3.8 习题与上机练习	91
第4章 ASP.NET的内置对象	94
4.1 HTTP 请求处理	94
4.1.1 Request 对象	94
4.1.2 Response 对象	98
4.1.3 Server 对象	101
4.2 状态信息保存	105
4.2.1 Application 对象	105
4.2.2 Session 对象	107
4.2.3 Cookie 对象	114
4.2.4 ViewState 对象	116
4.3 Trace 对象	118
4.4 ASP.NET运行配置文件	118
4.4.1 配置文件特点	119
4.4.2 配置文件结构	119
4.4.3 ASP.NET代码隐藏技术	120
4.5 习题和上机练习	120
第5章 ADO.NET数据库访问技术	123
5.1 ADO.NET体系结构	123
5.1.1 数据提供程序	123
5.1.2 DataSet 数据集	125

5.1.3 ADO.NET 类的组织	126
5.2 基于数据提供程序的数据库访问	126
5.2.1 数据库访问的一般方法	126
5.2.2 Connection 对象	128
5.2.3 Command 对象	134
5.2.4 DataReader 对象	144
5.2.5 DataAdapter 对象	146
5.3 DataSet 的架构	150
5.3.1 DataTable	150
5.3.2 DataView	152
5.4 习题和上机练习	156
第 6 章 ASP.NET 中数据绑定技术	158
6.1 数据绑定概述	158
6.1.1 数据绑定表达式	158
6.1.2 单值绑定	159
6.1.3 重复值绑定	160
6.2 数据源控件	162
6.2.1 数据源控件概述	162
6.2.2 使用 SqlDataSource 控件	166
6.3 富数据控件	170
6.3.1 GridView 控件	170
6.3.2 ListView 控件	187
6.3.3 DetailsView 控件和 FormView 控件	189
6.4 习题与上机练习	195
第 7 章 ASP.NET Web 服务	197
7.1 Web 服务概述	197
7.1.1 Web 服务的定义	197
7.1.2 Web 服务的基本特征	198
7.1.3 Web 服务的体系结构	198
7.1.4 Web 服务的协议栈	200
7.2 构建 ASP.NET Web 服务	201

7.2.1 创建 Web 服务	201
7.2.2 测试 Web 服务	203
7.2.3 发布 Web 服务	203
7.3 使用 Web 服务	205
7.3.1 生成服务代理类	205
7.3.2 添加 Web 引用	205
7.3.3 访问 Web 服务	206
7.4 习题与上机练习	207
第 8 章 JavaScript 脚本语言	209
8.1 JavaScript 概述	209
8.1.1 JavaScript 特性	209
8.1.2 网页嵌入 JavaScript	210
8.2 JavaScript 基本语法	212
8.2.1 JavaScript 规范与格式	212
8.2.2 JavaScript 注释	212
8.2.3 基本输出与交互方法	213
8.2.4 数据类型和变量	215
8.2.5 运算符和表达式	216
8.2.6 流程控制	217
8.3 函数	221
8.3.1 如何定义函数	222
8.3.2 函数的调用	222
8.3.3 变量的作用域和返回值	224
8.3.4 异常处理	224
8.3.5 JavaScript 事件处理	225
8.4 JavaScript 对象	227
8.4.1 基本对象	228
8.4.2 浏览器对象	233
8.4.3 HTML DOM 对象	237
8.5 JavaScript 编程实例	242
8.5.1 表单提交验证	242
8.5.2 时间计算程序	244

8.5.3 向表格中动态添加行	245
8.6 习题和上机练习	247
第9章 AJAX简介	250
9.1 AJAX概述	250
9.2 AJAX基本工作原理	251
9.2.1 AJAX实现步骤	251
9.2.2 AJAX异步调用示例	254
9.2.3 HTTP异步请求实例	257
9.3 DOM简介	261
9.4 Microsoft AJAX	265
9.4.1 概述	265
9.4.2 使用 UpdatePanel 控件实现页面局部刷新	266
9.4.3 使用 UpdateProgress 控件显示更新进度	268
9.4.4 使用 Timer 控件实现定时刷新	269
第10章 实例开发:物流管理系统	271
10.1 需求分析	271
10.1.1 系统设计目的	271
10.1.2 模块功能简介	271
10.1.3 功能要求	274
10.2 业务流程	278
10.2.1 业务流程图	278
10.2.2 框架构成	278
10.3 系统设计	279
10.4 系统数据字典	283
10.5 函数 API 接口说明	283
10.6 IE 浏览器配置	291
10.7 系统安装及配置	296
10.7.1 操作系统以及环境	296
10.7.2 B/S 站点部署	296
参考文献	301

第 1 章 ASP.NET 基础

ASP.NET 是在 Microsoft .NET 框架基础上构建的、可提供构建企业级 Web 应用程序所需服务的一个 Web 平台。ASP.NET 与 Microsoft .NET 框架平台紧密结合是 ASP.NET 的最大特点。本章主要介绍 .NET 应用开发架构、Microsoft .NET 框架、构建 ASP.NET 运行和开发环境，以及如何编写 ASP.NET 应用程序。

1.1 .NET 应用开发架构

1.1.1 .NET 的设计目标

.NET 最终目的就是让用户在任何地方、任何时间，以及利用任何设备都能访问所需的信息、文件和程序。用户不需知道这些文件放在什么地方，只需要发出请求，然后只管接收就可以了，而所有后台的复杂性是完全屏蔽起来的。

.NET 致力将手机、浏览器和门户应用程序集成到一起，形成一个统一的开发环境，结构如图 1-1 所示。

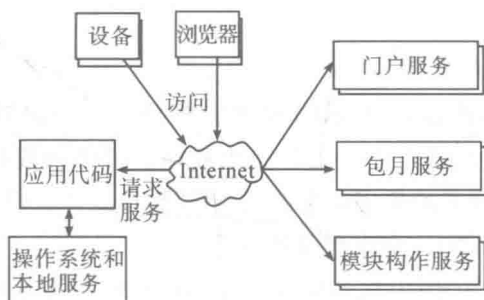


图 1-1 .NET 的战略

1.1.2 Microsoft .NET 框架

.NET 是一种面向网络、支持各种用户终端的开发平台环境，其目标是搭建新一代的因特网计算平台，解决网站间的协同合作问题，从而最大限度地获取信息。在 .NET 平台上，不同网站间通过相关协定联系在一起，形成自动交流、协同工作，提供全面的服务。

Microsoft .NET 框架(.NET Framework)是一个集成在 Windows 中的组件，支持生成和运行下一代应用程序与 XML Web Services，如图 1-2 所示。



图 1-2 .NET Framework

.NET Framework 具有两个主要组件：公共语言运行时(common language runtime, CLR)和.NET Framework 类库(.NET Framework class library)。

1. 公共语言运行时(CLR)

CLR 是.NET Framework 的基础。它为执行.NET脚本语言编写的代码提供了一个运行环境。CLR 管理.NET代码的执行，提供内存管理、线程管理、代码执行、安全验证、远程处理等服务，并保证应用和底层操作系统之间必要的分离。同时，CLR 使开发人员可以调试和进行异常处理。要执行这些任务，需要遵循公共语言规范(common language specification, CLS)。CLS 描述了运行库能够支持的数据类型的子集。

在 CLR 监视之下运行的程序属于受控代码，也叫托管代码(managed codes)。而不在 CLR 监视之下，直接在裸机上运行的应用或者组件叫非托管代码(unmanaged codes)。

2. .NET Framework 类库

.NET Framework 类库是一个与 CLR 紧密集成的、面向对象的、可重用的类型集合。.NET Framework 类型是生成 .NET 应用程序、组件和控件的基础。.NET Framework 可以理解为一系列技术的集成，它包括.NET脚本语言、CLS、.NET Framework 类库、CLR、Visual Studio .NET集成开发环境等。

.NET Framework 是用于构建、开发以及运行 Web 应用程序和 Web Service 的公共环境。它主要由三部分组成：编程语言(programming languages)、服务器端和客户端技术(server technologies and client technologies)、开发环境(development environments)。

1) 编程语言

- C# (读作 C sharp)

C# 是一种简洁、类型安全的、面向对象的语言，它是 Microsoft 公司专门为.NET量身定做、为生成在 .NET Framework 上运行的应用程序而设计的。C# 从 C 和 C++ 衍生而来，更像 Java。使用 C# 可以创建 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 Basic .NET (VB .NET)

VB .NET 是从 Visual Basic (VB) 语言演变而来，面向.NET Framework、能生成类型安全和面向对象应用程序的一种语言。VB .NET 是 Visual Studio .NET 的一部分，是一

套完整的、可生成企业级 Web 应用程序的开发工具。

- J# (读作 J sharp)

J# 是一种供 Java 程序员构建在 .NET Framework 上运行的应用程序和服务的语言。

2) 服务器端和客户端技术

- ASP.NET

ASP.NET 是建立在 .NET Framework 之上, 利用 CLR 在服务器端为用户提供建立强大的企业级 Web 应用服务的编程框架。ASP.NET 主要包括 Web Form 和 Web Service 两种编程模型。前者提供建立功能强大的、基于 Form 的可编程 Web 页面。后者提供在异构网络环境下获取远程服务、连接远程设备、交互远程应用的编程界面。

- Windows Forms

Windows Forms 是 .NET Framework 的智能客户端组件。利用 Visual Studio 之类的开发环境, 使用 Windows Forms 可以创建应用程序和用户界面。Windows Forms 应用程序是基于 System.Windows.Forms 命名空间中的类、通过在窗体上放置控件并对用户操作(如鼠标单击)进行响应来构建的。

- Compact Framework

Compact Framework 是为在移动设备和嵌入式设备上运行而设计的。它包含 .NET Framework 中类库的子集, 同时还包含一些专有类。

3) 开发环境

- Visual Studio

Visual Studio 是一个完整的集成开发环境 (integrated development environment, IDE), 用于生成 ASP.NET Web 应用程序、XML Web services、桌面应用程序和移动应用程序。VB.NET、Visual C++.NET、Visual C#.NET 和 Visual J#.NET 全都使用相同的 IDE, 该环境允许它们共享工具并有助于创建混合语言解决方案。

- Visual Web Developer

Visual Web Developer 是一个功能齐备的开发环境, 可用于创建 ASP.NET Web 应用程序。Visual Web Developer 提供网页设计、代码编辑、测试和调试, 以及将 Web 应用程序部署到承载服务器等功能。

.NET 框架提供统一的编程模式: 不论是 VB.NET、ASP.NET、C#、Jscript.NET 还是 .NET Web 服务都是用一样的 API。

1.2 ASP.NET 概述

ASP.NET 又称 ASP+, 它不仅仅是 ASP 的简单升级, 而是 Microsoft 推出的新一代脚本语言。ASP.NET 是 .NET 的一部分, ASP.NET 吸收了 ASP 以前版本的最大优点并参照 Java、VB 语言的开发优势加入了许多新的特色, 同时也修正了以前 ASP 版本的运行错误。

ASP.NET 是建立在 .NET 框架基础上的完全面向对象的系统。NET 框架平台给网站提供了全方位的支持, 包括强大的类库、多方面服务的支持、多种语言进行开发、跨平

台支持、充分的安全保障能力等。ASP.NET程序采用 Visual Studio .NET环境进行开发,支持 WYSIWYG(所见即所得)、拖放控件和自动部署等功能。

概括起来,ASP.NET与 ASP 的区别主要体现在以下几个方面。

1. 执行效率

ASP 是解释执行的,ASP.NET是编译执行的。ASP.NET采用编译后运行的方式,执行效率大幅提高。它将程序在第一次运行时编译成 DLL 文件,以后直接执行 DLL 文件,这样速度就变得很快。因此,ASP 在执行效率上大大低于实现同样功能的 ASP.NET。

2. 可重用性

ASP 将 HTML 代码和程序混在一起,代码的模块化和重用性低;ASP.NET将程序与 HTML 分开,实现了代码分离,程序结构清晰,代码的可重用程度高。ASP.NET可以向目标服务器直接复制组建,当需要更新时,重新复制一个即可。

3. 代码量

ASP 所有功能都要编写代码,而 ASP.NET提供了 Web 服务器控件,代码量大大降低。

ASP.NET与现存的 ASP 保持语法兼容,实际上我们可将 ASP 源文件扩展名.asp改为.aspx,然后配置在支持 ASP.NET运行的 IIS(Internet Information Services, Internet 信息服务)服务器的 Web 目录下,即可获得 ASP.NET运行时的优越性能。

1.2.1 ASP.NET的脚本语言

ASP.NET目前能支持 3 种语言: C#、Visual Basic .NET和 Jscript .NET。

C#是微软公司为.NET量身定做的编程语言,它与.NET有着密不可分的关系。C#的类型就是.NET框架所提供的类型,C#没有类库,使用.NET框架所提供的类库。另外,类型安全检查、结构化异常处理也都是交给 CLR 处理的。因此,C#是最适合开发.NET应用的编程语言。

Visual Basic .NET是在现有 Visual Basic 6.0 基础上的一次重大的飞跃。Visual Basic .NET对 Visual Basic 进行了重塑,使得 Visual Basic .NET比 Visual Basic 6.0 更易用、更强大,同时加入了过去只有使用 C++语言才能实现的某些系统资源的访问能力,最重要的是 Visual Basic .NET完全支持面向对象技术。

Jscript .NET是在现有的 Jscript 语言基础上做了彻底的修改,加入了面向对象特性。

1.2.2 ASP.NET的工作原理

ASP.NET的工作原理如下所述,参见图 1-3。



图 1-3 首次请求 ASP.NET 页面的处理过程

(1) Web 浏览器发送一个 HTTP 请求到 Web 服务器，要求访问一个 Web 网页。

(2) Web 服务器分析这个 HTTP 请求，定位所请求的 Web 网页的位置。

(3) 如果请求的网页是一个 HTML 文件，则服务器直接返回该文件。如果请求的网页是个 ASP.NET 文件，那么 IIS 就把该文件传送到 `aspnet_isapi.dll` 进行处理，后者把 ASP.NET 代码提交给 CLR。若是首次请求这个 ASP.NET 文件，就由 CLR 编译并执行，得到纯 HTML 结果；若是已经执行过这个文件，那么就直接执行编译好的程序并得到纯 HTML 结果。

(4) 最后把从 (3) 中得到的 HTML 文件传回浏览器作为 HTTP 响应。浏览器收到这个响应之后，就可以显示 Web 网页。

1.3 建立 ASP.NET 的运行和开发环境

要运行 Web 程序，必须首先建立一个 Web 服务器，然后从任一台 Web 浏览器访问该服务器上的 Web 程序。建立 ASP.NET 的运行环境，需要安装 Web 服务器 IIS 和 .NET Framework。

要编写和调试 ASP.NET 程序，通常选择 Visual Studio 集成开发环境。本书以当前主流配置的 Visual Studio 2010 和 SQL Server 2008 为主要开发环境进行讲解。

1.3.1 安装和配置 IIS 服务器

IIS 是 ASP.NET 唯一可以使用的 Web 服务器，目前常用的版本是 IIS 6.0 或 IIS 7.0。下面以 Windows XP 为例，简述 IIS 6.0 的安装和配置步骤。

1. 安装 IIS

(1) 选择“控制面板→添加/删除程序→添加/删除 Windows 组件”。

(2) 在“Windows 组件向导”窗口中选中“Internet 信息服务(IIS)”，单击“下一步”按钮，按照提示插入 Windows 安装光盘后，将自动安装 IIS。

(3) 安装结束后，在浏览器地址栏中输入：`http://127.0.0.1` 或 `http://localhost`，如果出现 IIS 的信息页面，则表示 IIS 安装成功。此时，通常会在硬盘 C 上自动创建文件夹 `C:\Inetpub`，这是 IIS 的默认目录。`C:\Inetpub\wwwroot` 是默认的 Web 主页的地址。

2. 设置虚拟目录

假设一个 ASP.NET 文件 hello.aspx 存放在服务器上的某一目录 D:\myWebSite 里, 如例 1-1 所示。

[例 1-1] 一个简单的 ASP.NET 程序(test1_1.aspx)。

```
<%@page language="C#" %>
<html>
<body>
<font color="red">
<% Response.Write("hello world.");%>
</font>
</body>
</html>
```

下面介绍如何配置 IIS, 以及运行该程序的方法。

(1) 选择“控制面板→管理工具→Internet 服务管理器”, 进入“Internet 信息服务”主窗口。

(2) 展开“默认网站”, 选择“新建→虚拟目录”, 启动“虚拟目录创建向导”程序, 输入别名“mysite”, 并指向目录“D:\myWebSite”, 最后选择正确的访问权限。至此, 建立了一个别名为 mysite 指向 D:\myWebSite 的虚拟目录。

然后, 在浏览器中输入以下地址, 就可以看到图 1-4 所示的结果:

http://localhost/mysite/hello.aspx

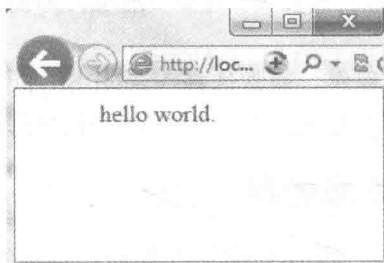


图 1-4 测试和运行 ASP.NET 程序

1.3.2 安装 Visual Studio 开发工具

要安装 Visual Studio 开发环境, 必须首先安装 .NET Framework, 然后再安装 Visual Studio 工具包。本书选择安装 Microsoft Visual Studio 2010 开发工具包。

Microsoft Visual Studio 2010 安装程序通常自带 .NET Framework 4.0, 后者也可以从 Microsoft 公司的官方网站 www.microsoft.com/downloads 下载。

Visual Studio 2010 目前有 5 个版本: 专业版、高级版、旗舰版、测试专业版和速成版。

基于普遍性的考虑, 本书将安装 Visual Studio 2010 Ultimate 旗舰版, 简称 VS 2010

Ultimate。它可以通过下载一个 ISO 镜像文件进行安装。

VS 2010 Ultimate 包括以下工具。

- Visual Basic
- Visual C++
- Visual C#
- Visual F#
- Visual Web Developer

本书选择安装 Visual C# 和 Visual Web Developer。

安装完成后,首次启动 VS 2010 Ultimate,选择默认的环境设置,然后进入如图 1-5 所示的起始页面。



图 1-5 VS 2010 Ultimate 的起始页

通常,开发 ASP.NET 程序的第一步就是创建一个新网站。

如果选择“文件→新建网站”,打开一个“新建网站”对话框,进入图 1-6 所示的画面。选择“ASP.NET 网站”,在文本框中输入网站的文件夹位置,然后单击“确定”按钮,Visual Studio 将创建一个网站项目,并自动生成一个 Default.aspx 的网站默认主页。

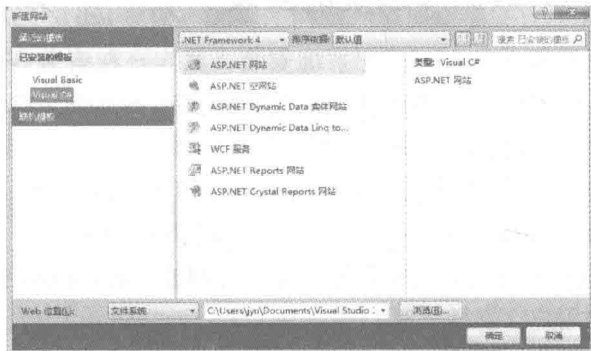


图 1-6 “新建网站”对话框

从“网站”菜单中选择“添加新项”,然后选择“Web 窗体”,则添加一个 ASP.NET 页面。单击“添加”按钮,进入图 1-7 所示的 Visual Studio 集成开发环境(IDE)主