

第一步 ✓

留在你记忆深处的第一步

Hello, World


ASP.NET 第一步

——基于 C# 和 ASP.NET 2.0

朱晔 著

《第一步》系列六大特点：

- 对概念的地道讲解
- 对疑难点的透彻剖析
- 贯穿全书的大量短小精悍的 Demo 级例子
- 二至五个完整的项目实例
- 充分考虑到初学者在学习和实际开发中会遇到的各种困难并及时给出相应的解答
- 为读者提供极具实用性的一揽子解决方案

 附源代码光盘



原创经典
清华大学出版社



原创经典编程大系留在你记忆深处的第一步

ASP.NET 第一步

——基于 C#和 ASP.NET 2.0

朱 晔 著

清华大学出版社
北 京

内 容 简 介

本书以清晰的条理、精炼的例子把初学者带入 ASP.NET 2.0 的世界。

全书共分四个部分。“基础篇”在涵盖了基础知识的同时以留言簿的示例让初学者快速入门，迅速激发学习的热情。“技术篇”在覆盖常用技术的同时也涉及到了 ASP.NET 的一些底层实现。“提高篇”通过讲解进阶技术，帮助读者编写出更好的代码，开发出更强大的 ASP.NET 应用程序。“扩展篇”介绍了 ASP.NET 2.0 常见的企业级开发构架和技术，为读者以后的深入学习作好了铺垫。

本书具有以下四大特点：

- 紧凑精炼的篇幅：内容覆盖广且每个章节都短小精悍。
- 语言贴近初学者：摒弃难以理解的 MSDN 的定义和示例，对疑难的知识点都有贴切的比喻和讲解。
- 独特的章节安排：根据切入点和疑难点来安排章节，确保初学者在不知不觉中顺畅阅读。
- 示例式教学方式：倡导从示例入手的教学方式，先手把手让您获取成果再分析为什么来增加读者的信心和兴趣。

本书封面贴有清华大学出版社防伪标签，无标签者不得销售。

版权所有，侵权必究。侵权举报电话：010-62782989 13501256678 13801310933

图书在版编目 (CIP) 数据

ASP.NET 第一步：基于 C#和 ASP.NET 2.0 / 朱晔著. —北京：清华大学出版社，2007.7
(第一步)

ISBN 978-7-302-15223-1

I. A… II. 朱… III. 主页制作—程序设计 IV. TP393.092

中国版本图书馆 CIP 数据核字 (2007) 第 071273 号

责任编辑：陈 冰

封面设计：陈 冰

责任校对：张 剑

责任印制：王秀菊

出版发行：清华大学出版社 地 址：北京清华大学学研大厦 A 座

<http://www.tup.com.cn> 邮 编：100084

c-service@tup.tsinghua.edu.cn

社 总 机：010-62770175 邮购热线：010-62786544

投稿咨询：010-62772015 客户服务：010-62776969

印 刷 者：北京嘉实印刷有限公司

装 订 者：三河市金元印装有限公司

经 销：全国新华书店

开 本：203×260 印 张：38.5 字 数：1051 千字

版 次：2007 年 7 月第 1 版 印 次：2007 年 7 月第 1 次印刷

印 数：1~5000

定 价：69.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题，请与清华大学出版社出版部联系
调换。联系电话：(010)62770177 转 3103 产品编号：023791-01



朱晔 微软最有价值专家 (2006年、2007年 ASP.NET 领域 MVP), CSDN 社区 ASP.NET 版主, 博客园排名 20。自 2000 年开始致力于 Web 开发的研究和推广。爱好 .NET 技术, 长期活动于 CSDN 社区和博客园, 并在业余时间参与翻译 .NET 书籍 (近期译作《Pro C# 2005 and the .NET 2.0 Platform, Third Edition》)。现就职于上海麦格特尔网络科技有限公司。

欢迎读者通过个人博客 (<http://www.cnblogs.com/LoveCherry/>) 或 MSN (邮件): yzhu@live.com 和我交流、探讨技术。

前 言

本书是一本 ASP.NET 2.0 的入门书籍，针对未接触过 ASP.NET 开发（但是读者需要有一定的计算机基础和其他编程语言的基础）或者有一些 ASP.NET 1.1 开发经验的编程爱好者自学使用。笔者也是从一个初学者走过来的，从 ASP、PHP 到 ASP.NET 1.1/2.0 一路上靠的都是自学，我深深体会到对于任何一门技术来说往往是两头难，也就是入门难，精通难。对于学习 Web 开发来说更是如此，Web 开发不是一门语言，而是多种技术、多个语言的集合。通常，要构建一个大型的网站需要如下的一些专业人员。

- 网络工程师。负责服务器和网络设备硬件软件的部署和调试。
- 网络程序员。负责网站程序部分的开发。
- 美术设计师。负责网站的美术设计，风格设计。
- 数据库管理员。负责数据库编程和优化。

当然，如果是制作一个小网站的话这些职务就只能由一个人承担了。

- 要学会服务器、交换机等设备的安装，服务器操作系统的安装配置。
- 要有一定的美术设计能力，会使用 Photoshop 等软件进行设计。
- 要掌握一定的数据库知识，学会 Sql Server 等数据库的使用和优化。
- 要掌握一门 Web 编程语言，比如会使用 ASP.NET 开发网站。

一名合格的网络程序员不但需要了解自己领域的一些专业技术，而且很多时候还需要充当半个网络工程师、半个美术设计师和半个数据库管理员。想到这些，初学者就害怕了。入门虽说不易，但一个好的切入点能让你快速入门。为了能让你入门，本书的策略是从实际的东西到抽象的东西，从必要的技术到辅助的技术，在入门以前不谈实质。笔者承认从一门技术的底层开始学习能有一个扎实的基础，但是这样的入门方式实在是太漫长了，不如从实际的东西开始，先尝试做出一些东西，然后再去探究原因，这种曲折式的学习方法更适合初学者。

市面上的技术书籍大多分如下两种。

- 以知识点划分章节。70%的书都是这样。这种类型的书，优点是对一门技术有一个系统的介绍，缺点是切入点不明确，初学者往往需要看几遍才能入门，而入门后才发现书的后 10 章（相对深入的一些内容）没有什么用处。
- 以大型实例划分章节。整个书就是 2~3 个大型的实例，满书的源代码，而且源代码的质量往往不怎么高。初学者看了以后不能入门，而入门的人也不再需要看这些书了，因为这些所谓的实例自己都可以做出来。

本书的特点：

- 笔者把知识点进行了充分打散，在保证知识点相对集中的情况下又把知识点按照难易程度进行分配。而且，对于一些华而不实的知识点本书都是一笔带过，笔者认为一本入门书籍不能把读者带入误区，初学者的接受能力有限，一定要让他们首先了解最重要的知识点而不是一些用得比较少而又比较难的知识点。

- 本书的示例也非常有特点，小知识点的示例简单精炼，构架方面知识点的示例实际而又有效（比如在介绍数据访问、无刷新的时候都是以同一个留言簿的程序为例子），在介绍 ASP.NET 核心知识的时候又使用了贯穿的论坛作为大例。根据不同的需求来使用不同规模的例子，既不浪费篇幅又能全方位解析知识点。

笔者始终认为初学者最需要的不是一本大全或者宝典，他们最需要以下两样东西。

- 基本而实用的例子来建立信心。
- 开发的经验和心得避免今后走弯路，这是初学者最欠缺的。

因此，笔者认为三种书是有用的。

- 纯粹给初学者看的书。用一步一步手把手的方式教大家在实际中入门，看了以后既能做出一定的东西又知道了一些原理。
- 包含了实际经验的书。体现出作者多年的实战经验，能让入门的人迅速提升功力，在构架和设计上得到提高。
- 纯粹给高手看的书。只剖析底层的实现，不涉及应用上的一些东西。

本书既有大量的向导实例，又穿插了很多开发的经验。能让初学者每一次阅读本书都有收获，如果你能认真阅读三遍，那么你肯定入门了。

很多初学者会问，Java 和 .NET 我学哪一种？哪个比较有前途？我知道，他们是在担心学的技术很快会落伍，担心学的技术没有前景。常说选择是痛苦的，但我认为根本没有必要选择。编程语言之间都是相通的，对一门编程语言达到精通的人来说，转向另外一门语言的学习和开发不会发生本质转变。其实，编程技术和武侠小说里面的武功有很多相像之处。武功分内功和招式两部分，所谓内功是习武之人多年积累下来的武术功底，内力强大的人无论学哪一种武功都能快速掌握并发挥出威力。相反，招式只是应用上的东西，如果只学招式不练内功那么再好的武功都仅仅是花拳绣腿。初学 .NET 的时候可能你需要花 1 年的时间入门，做了 5 年 .NET 后你会发现即使项目彻底转向 Java，你也只需要 1 个月的时间就能入门。如果你做了 5 年 .NET 而转向另外一种技术的学习和开发，这 5 年绝不是白费的。因此，在入门以后你积累的一些经验就好似内功积淀了下来。举例来说，你知道使用 .NET 做设计的时候，哪里会是性能上的瓶颈，即使转向 Java，这个经验也能给你提供很大的帮助。

读者可能会关心何种学习方式才是合理的，笔者结合自己的经验谈一谈。每个人的入门方式可能都不一样。

- 有人是因为一个无奈的项目入门，客户一定要使用 ASP.NET。而我只会 ASP，只能硬着头皮上。
- 有人是因为一门课程入门，学校开了一门 ASP.NET 的课程。在考试结束后，突然发现还挺有趣的。
- 有人是因为兴趣入门，这种没有任何压力的入门方式是最难的，因为不容易找到突破口。

对于初学者笔者推荐如下的入门方式（针对 Web 开发来说）。

1. 了解一些最基本的概念和技术。
2. 构建开发平台。安装配置开发环境。
3. 周边核心技术的学习。开发语言、开发脚本。
4. 快速入门。调试成功一个完整的实例，比如使用书上的代码制作一个留言簿，然后琢磨每一个语句的作用。
5. 不看书自己尝试做一个最简单的东西。
6. 系统学习一些具有实践价值的知识，结合自己的练习。

7. 看一些别人写的代码，在理解代码的基础上进行总结。

8. 至此你已经入门了，做两个具有一定意义的中型项目（比如论坛和图书馆系统），能让你从入门到熟练。

在整个学习过程中，你还要学会使用一些工具来解答你的问题。因为不是每一个初学者的周围都会有高手回答自己的问题。我们要学会自己解决问题。

- 最好的搜索——Google。一定要学会搜索，全世界这么多人在学习 ASP.NET，如果你遇到问题，这个问题 99% 别人也遇到过。现在网络这么发达，无数人在 BLOG、论坛分享自己的解决方案。相信 50% 的问题，Google 能告诉你答案。
- 最好的参考——MSDN。不知道怎么用一个类、方法不要紧，MSDN 对所有的类型都会有最准确而详细的解释。不用去买什么参考书籍，MSDN 上的内容不但时效性强而且准确。
- 最好的老师——论坛。实在遇到了一个奇怪的问题又不知道怎么搜索，你可以在论坛上求助（比如 CSDN 社区），网友会热心解答，尝试在多个技术论坛提出问题得到解答的概率会高很多。

本书既在广度上覆盖了 Web 开发的方方面面（包括最新的、流行的技术）又不放弃很多细节。那么，怎么样阅读本书才能达到最大的效果呢？

- 如果你从未接触过 Web 开发。可以先看 1、2 两章后直接看第 6 章快速入门，建立信心。然后再从头开始看，在看的过程中遇到看不懂的直接跳过去，比如在看第 9 章的时候看到最后一节看不懂，那就请直接看第 10 章，看完第 10 章再回头看第 9 章。你会发现每一次重复阅读以前看过的章节都会有新的收获。
- 如果你接触过 Web 开发，但未接触过 .NET 开发。可以重点看第 4、5 章，多做一些小例子，从熟悉 C# 语言开始，然后再重点阅读与 ASP.NET 特性相关的一些章节（第 8~12 章）。
- 如果你曾接触过 ASP.NET 2.0 以前版本的开发。可以粗略阅读技术篇（看看 2.0 多了哪些特性）然后重点阅读提高篇，让自己的知识在广度上有一个扩展。
- 如果你已经有 ASP.NET 2.0 的开发经验。阅读扩展篇能让你在构架上有初步的认识。其他章节你也可以通读，补上自己基础知识的漏洞。

最后要感谢我的朋友们（特别是卓见、朱磊、苏宁、马国鹏、沈绍虞以及龚耀武）在整个过程中给我的很多帮助和意见。

一句话，希望本书对读者有用。当然，由于时间仓促，书中难免有一些错误。在阅读的过程中有任何问题，都可以直接和笔者进行联系：yzhu@live.com。

业界评语

认识 LoveCherry 是在 CSDN 的 ASP.NET 版，那时候他正所向无敌、准确无误地回答每一个人的问题，很多直接点名找他帮忙的帖子非常显眼，想不认识他都难。他回答的问题涉及面很广，小到实现的细节大到一个解决方案，他都一丝不苟，认认真真地帮助解决到底，让人不得不佩服他对技术了解的深度和广度，对待知识的严谨、认真和执着。

我们聊天的时候，他说过从提问的问题来看，很多初学者都会有一些误区，怎样才能让大家少走弯路，在有限的时间内学到更多的知识呢？我理解他的想法，我想他的经验和心得对提问的人来说是重要的，对更多的 ASP.NET 初学者和开发者来说也是重要的，他也许应该把自己掌握的知识、积累的经验写下来，分享给更多的人。后来听说他要出书的时候，真的非常高兴，替他和能够分享这些知识的人感到高兴。之后的半年内，我陆续地读到本书的一些章节，只能用“惊喜”来形容了。

这本书绝对可以成为一个好学者的宝贝，和其他技术类图书不同的是，LoveCherry 完全是站在分享自己心得和经验的角由浅入深，循序渐进地讲解，以示例贯穿全书，对其中的每一个知识点都分析得清晰而透彻。

我相信这本书的出版，对于想要学习和正在学习 ASP.NET 的朋友绝对是一份非常好的礼物。

——于淼（网名：大象/CMIC）

微软最有价值专家（MVP）

美国系统技术公司（ASTI）全资子公司 宏元讯通首席构架师

ASP.NET 是一种建立动态 Web 应用的技术，经过多年的发展已经渐入佳境。相比其他的 Web 技术来说，ASP.NET 能更快速、更容易地建立灵活、安全、稳定的 Web 应用。本书内容详实、浅显易懂，无论您是初学者还是开发高手，都能从中获益。如果您想分享作者 LoveCherry 丰富的 ASP.NET 开发经验，就打开这本杰作吧！

——张大磊 微软开发技术经理

目 录

第 I 部分 基础篇

第 1 章 Web 编程概述	2	3.2.5 四则运算的例子	27
1.1 浏览器/服务器 (B/S) 模型	2	3.3 字符串操作	28
1.1.1 客户端/服务器 (C/S) 模型的问题	2	3.4 流程控制	30
1.1.2 浏览器/服务器 (B/S) 模型的优劣	2	3.4.1 流程控制概述	30
1.2 Web 服务器与 Web 请求	3	3.4.2 选择语句	31
1.3 Web 开发技术的发展	3	3.4.3 迭代语句	33
1.3.1 客户端技术的发展	3	3.4.4 冒泡排序的例子	37
1.3.2 服务端技术的发展	5	3.4.5 字符串与整数互转的例子	38
1.3.3 J2EE vs 微软.NET	6	3.5 了解面向对象编程	40
1.4 ASP.NET 技术	6	3.5.1 面向对象概述	40
1.4.1 彻底的颠覆——ASP.NET 与 ASP	6	3.5.2 冒泡排序类	40
1.4.2 更强更完善——ASP.NET 2.0 与 1.1	7	3.5.3 类和成员	46
1.5 其他基础知识	8	3.5.4 值类型和引用类型	48
1.6 回顾与总结	8	3.5.5 汽车类的例子	50
第 2 章 建立开发环境	9	3.6 回顾与总结	53
2.1 安装 IIS	9	第 4 章 强大的.NET 类库	55
2.2 安装.NET 框架与集成开发环境	10	4.1 常用命名空间	55
2.3 熟悉 IIS	12	4.1.1 基础命名空间	55
2.3.1 注册 ASP.NET 2.0	12	4.1.2 图形命名空间	55
2.3.2 了解 IIS 的设置	13	4.1.3 数据命名空间	56
2.3.3 配置虚拟目录	15	4.1.4 Web 命名空间	56
2.4 熟悉集成开发环境	16	4.1.5 框架服务命名空间	56
2.5 回顾与总结	17	4.1.6 安全性命名空间	57
第 3 章 C#编程基础	18	4.1.7 网络命名空间	57
3.1 创建你的第一个 C#程序	18	4.1.8 配置命名空间	57
3.2 变量和运算	20	4.1.9 本地化命名空间	58
3.2.1 单值类型	20	4.2 集合类型	58
3.2.2 类型转换	21	4.2.1 System.Collections 命名空间概述	58
3.2.3 枚举与数组类型	23	4.2.2 使用 ArrayList 类型	58
3.2.4 运算操作	24	4.2.3 使用 Hashtable 类型	60
		4.2.4 使用 SortedList 类型	62
		4.2.5 使用 Stack 类型	64

4.2.6 泛型集合	68	5.4.1 存储过程与参数	99
4.3 磁盘操作	69	5.4.2 存储过程与事务	102
4.3.1 System.IO 命名空间概述	69	5.4.3 声明变量与循环	103
4.3.2 使用 DriveInfo 类型读取 驱动器信息	70	5.5 回顾与总结	104
4.3.3 使用 Directory 和 DirectoryInfo 类型操作目录	70	第 6 章 快速入门——简易留言簿	105
4.3.4 使用 File 和 FileInfo 类型 操作文件	72	6.1 编码前的准备	105
4.3.5 使用 StreamReader 和 StreamWriter 类型读写文件	73	6.1.1 回顾基本概念	105
4.4 反射	75	6.1.2 设计数据结构	106
4.4.1 System.Reflection 命名空间概述	75	6.2 实现	106
4.4.2 反射对象的成员	78	6.2.1 创建数据库	106
4.4.3 反射对象的字段	78	6.2.2 实现发表留言功能	107
4.4.4 反射对象的属性	79	6.2.3 显示用户发表的留言	113
4.4.5 反射对象的方法	80	6.2.4 管理员操作	116
4.5 图形图像处理	80	6.3 回顾与总结	120
4.5.1 System.Drawing 命名空间概述	80	第 II 部分 技术篇	
4.5.2 生成验证码的例子	81	第 7 章 .NET 框架与 ASP.NET	122
4.5.3 生成缩略图的例子	83	7.1 .NET 框架体系结构概述	122
4.5.4 生成黑白图的例子	85	7.2 公共语言运行库 CLR	122
4.6 网络	86	7.2.1 公共语言运行库概述	122
4.6.1 从远程地址下载文件的例子	86	7.2.2 编译的过程和原理	123
4.6.2 Ping 操作的例子	87	7.2.3 CLR 的自动内存管理	124
4.7 回顾与总结	87	7.3 程序集	124
第 5 章 数据库基础	89	7.3.1 程序集概述	125
5.1 使用 IDE 连接 SQL Server	89	7.3.2 程序集的构成	125
5.2 创建数据库与表	90	7.3.3 构建程序集信息	126
5.3 SQL 查询	93	7.4 应用程序域	129
5.3.1 记录添加	94	7.4.1 应用程序域概述	129
5.3.2 记录修改	94	7.4.2 应用程序域编程	129
5.3.3 记录删除	95	7.5 ASP.NET 代码结构与编译	132
5.3.4 单表查询	95	7.5.1 ASP.NET 代码模型	132
5.3.5 多表查询	97	7.5.2 ASP.NET 2.0 的保留文件夹	135
5.3.6 聚合函数	98	7.5.3 ASP.NET 2.0 页面模型	136
5.3.7 分组查询	98	7.5.4 ASP.NET 2.0 编译模型	138
5.3.8 子查询	99	7.6 ASP.NET 请求和响应	142
5.4 存储过程	99	7.6.1 ASP.NET 应用程序的生命周期	142
		7.6.2 ASP.NET 处理管线	143
		7.6.3 HttpRequest 对象	145

7.6.4	HttpResponse 对象	147	8.5.5	必须字段验证控件 RequiredFieldValidator	186
7.6.5	访问记录程序	147	8.6	网络文件夹的例子	187
7.6.6	文件下载程序	148	8.6.1	网络文件夹的界面预览	187
7.6.7	ASP.NET 页面生命周期	149	8.6.2	网络文件夹的功能需求	188
7.7	ASP.NET 事件模型	152	8.6.3	网络文件夹的代码实现	188
7.7.1	ASP.NET 事件模型概述	152	8.7	回顾与总结	199
7.7.2	ASP.NET 服务器控件事件模型	153	第 9 章	ADO.NET 数据访问模型	200
7.8	ASP.NET 2.0 Provider 模型概述	156	9.1	ADO.NET 概述	200
7.8.1	Provider 模型概述	156	9.2	连接数据库	201
7.8.2	Provider 模型基本结构	157	9.2.1	连接字符串	201
7.8.3	ASP.NET 2.0 中的 Provider	158	9.2.2	使用 Connection 对象 连接数据库	202
7.9	回顾与总结	160	9.2.3	使用 SqlConnectionStringBuilder 构建连接字符串	206
第 8 章	服务器控件	162	9.2.4	枚举所有可用数据源	207
8.1	服务器控件概述	162	9.3	执行 SQL 语句	207
8.1.1	传统的解决方案	162	9.3.1	回顾论坛数据库	207
8.1.2	ASP.NET 的解决方案	163	9.3.2	使用 SqlCommand 执行 SQL 语句	208
8.1.3	使用 IDE 添加控件	163	9.3.3	SQL 注入带来的威胁	211
8.1.4	控件的属性	164	9.4	使用参数化 SQL 与存储过程 提高安全性	213
8.1.5	控件的标签	165	9.4.1	使用参数来防止 SQL 注入	213
8.1.6	特殊的服务器控件	166	9.4.2	创建存储过程	214
8.2	服务器控件的层次结构	167	9.4.3	使用 SqlCommand 对象执行 存储过程	215
8.2.1	控件的层次结构与基类	167	9.5	使用 DataReader 访问数据	216
8.2.2	使用程序访问控件	168	9.5.1	使用 DataReader 读取单记录集	216
8.2.3	使用智能感知熟悉控件	169	9.5.2	使用 DataReader 读取多记录集	218
8.2.4	Web 服务器控件层次结构	170	9.6	DataSet 数据容器	220
8.2.5	HTML 服务器控件层次结构	171	9.6.1	DataSet 对象的基本结构	220
8.3	服务器控件树	172	9.6.2	创建 DataSet	221
8.3.1	使用程序输出控件树	172	9.6.3	访问 DataSet	222
8.3.2	LiteralControl 控件	174	9.6.4	使用 DataRelation 实现父子表	224
8.4	服务器控件的事件	176	9.7	离线模式操作数据库	226
8.4.1	为控件添加事件	176	9.7.1	使用 DataAdapter 来获取数据	226
8.4.2	事件参数与自动回发	177	9.7.2	使用 DataAdapter 来更新数据	229
8.4.3	服务端事件与客户端事件	178	9.8	回顾与总结	233
8.5	服务器控件的使用	180			
8.5.1	下拉列表控件 DropDownList	181			
8.5.2	多选框列表控件 CheckBoxList	182			
8.5.3	文件上传控件 FileUpload	183			
8.5.4	多重视图控件 MultiView	185			

第 10 章 数据绑定与数据绑定控件234	11.3.1 CSS 样式概述.....286
10.1 数据绑定概述——简化数据访问和 实现页面与程序的分离.....234	11.3.2 使用 CSS 统一页面样式.....290
10.1.1 单值绑定.....234	11.3.3 创建与应用主题.....291
10.1.2 绑定表达式.....235	11.3.4 创建控件的皮肤.....293
10.2 重复值绑定.....237	11.3.5 切换网站主题.....295
10.2.1 各种类型的数据源.....237	11.4 用户控件.....299
10.2.2 支持数据源绑定的控件.....238	11.4.1 创建用户控件.....299
10.2.3 把数据源绑定到控件.....239	11.4.2 使用用户控件.....302
10.3 复杂数据绑定控件之 GridView.....242	11.4.3 以编程方式访问用户控件.....304
10.3.1 回顾论坛数据库结构.....242	11.5 本地化和资源——创建多语言网站.....304
10.3.2 把数据源绑定到 GridView.....243	11.5.1 生成本地资源.....304
10.3.3 GridView 的绑定列与模板列.....244	11.5.2 创建多语言资源文件.....307
10.3.4 GridView 的事件.....250	11.5.3 实现多语言切换.....308
10.3.5 对你的 GridView 进行美化.....256	11.5.4 以编程方式访问本地资源.....309
10.3.6 格式化 GridView.....261	11.5.5 创建和使用全局资源.....310
10.3.7 数据的排序与分页.....262	11.6 导航控件和站点地图.....311
10.4 复杂数据绑定控件之 DataList——在 一行中显示多条记录.....266	11.6.1 创建自动的导航路径.....311
10.4.1 DataList 的属性与模板.....266	11.6.2 多语言和多主题的导航.....313
10.4.2 DataList 的模板与事件.....268	11.7 回顾与总结.....315
10.5 复杂数据绑定控件之 Repeater——完成 论坛分类和版块管理.....271	第 12 章 状态管理——记录你网站访客的活动316
10.5.1 使用 Web.config 存储 连接字符串.....271	12.1 状态管理概述.....316
10.5.2 使用 Repeater 呈现父子 表数据.....272	12.1.1 状态管理与数据库.....316
10.6 回顾与总结.....278	12.1.2 状态管理的比较参数.....316
第 11 章 ASP.NET 应用程序统筹280	12.2 Cookie.....317
11.1 ASP.NET 应用程序统筹概述.....280	12.2.1 Cookie 概述.....317
11.1.1 大型系统的需求.....280	12.2.2 Cookie 的使用.....317
11.1.2 ASP.NET 的解决方案.....280	12.2.3 Cookie 总结.....320
11.2 母版页——构建具有统一外观的 应用程序.....281	12.3 Session.....320
11.2.1 创建母版页.....281	12.3.1 Session 概述.....320
11.2.2 创建基于母版页的 Web 窗体.....283	12.3.2 Session 的使用.....321
11.2.3 以编程方式访问母版页.....286	12.3.3 把 Session 存储在独立 的进程中.....322
11.3 为你的网站创建主题和皮肤.....286	12.3.4 把 Session 存储在数据库中.....324
	12.3.5 Session 的使用范围与 大小限制.....325
	12.3.6 Session 的生命周期.....327
	12.3.7 遍历与销毁 Session.....328
	12.3.8 Session 的常见问题与总结.....328
	12.4 Application.....329

12.4.1 全局应用程序类.....	329	14.3 HttpModule.....	384
12.4.2 使用 Application 统计 网站访问.....	331	14.3.1 Http 模块简介.....	384
12.4.3 Application 总结.....	332	14.3.2 创建简单的 Http 模块——一个 IP 过滤的例子.....	384
12.5 Cache.....	333	14.4 回顾与总结.....	386
12.5.1 Cache 概述.....	333	第 15 章 保护你的 ASP.NET 应用程序.....	387
12.5.2 Cache 性能与过期策略.....	333	15.1 ASP.NET 应用程序安全概述.....	387
12.5.3 Cache 的更新策略.....	337	15.1.1 安全相关的名词.....	387
12.5.4 Cache 总结.....	339	15.1.2 ASP.NET 安全结构.....	387
12.6 隐藏域/ViewState/ControlState——保存 数据的另一个场所.....	339	15.1.3 身份验证方式.....	388
12.6.1 使用隐藏域.....	339	15.2 用户账户模拟.....	388
12.6.2 使用 ViewState.....	340	15.2.1 一个模拟请求用户的例子.....	388
12.6.3 ViewState 的安全与性能.....	340	15.2.2 模拟某一个用户.....	392
12.6.4 ControlState 概述.....	343	15.2.3 使用编程方式进行临时模拟.....	392
12.6.5 总结.....	343	15.3 基于 Windows 的身份验证.....	394
12.7 以人为本的 Profile.....	343	15.3.1 概述.....	394
12.7.1 使用 Profile 制作个性化页面.....	343	15.3.2 配置 IIS 安全.....	394
12.7.2 Profile 总结.....	348	15.3.3 配置 Windows 安全.....	395
12.8 其他.....	349	15.3.4 配置 Windows 身份验证.....	397
12.8.1 QueryString.....	349	15.3.5 配置 Windows 授权.....	397
12.8.2 跨页提交.....	349	15.3.6 实现自定义角色.....	398
12.9 完善论坛.....	350	15.3.7 获取用户信息.....	399
12.10 回顾与总结.....	355	15.4 基于表单的身份验证.....	399
第 13 章 强类型 DataSet.....	357	15.4.1 概述.....	399
13.1 强类型 DataSet 概述.....	357	15.4.2 启用表单身份验证.....	399
13.2 使用强类型 DataSet 进行数据访问.....	357	15.4.3 配置表单身份验证.....	400
13.3 论坛程序总结.....	368	15.4.4 配置表单授权——一个限制 匿名下载的例子.....	402
13.4 回顾与总结.....	371	15.4.5 登录与注销.....	403
第 III 部分 提高篇		15.4.6 使用 Web.config 进行 身份验证.....	405
第 14 章 请求上下文.....	374	15.4.7 获取用户信息.....	406
14.1 HTTP 上下文简介.....	374	15.5 ASP.NET 2.0 的成员资格和 角色管理器.....	406
14.2 HttpHandler.....	375	15.5.1 成员资格概述.....	406
14.2.1 Http 处理程序简介.....	375	15.5.2 成员资格的配置.....	407
14.2.2 自定义 HTTP 处理程序——实现 图片防盗链的功能.....	375	15.5.3 角色管理器概述.....	409
14.2.3 IHttpHandlerFactory——实现 文件下载管理.....	379	15.5.4 角色管理器的配置.....	410
		15.5.5 安全性相关的控件.....	411
		15.5.6 ASP.NET 网站管理工具.....	412

15.5.7 成员资格和角色管理器 API 编程.....413	第 17 章 Ajax 无刷新技术.....450
15.6 安全性相关的控件.....413	17.1 Ajax 概述.....450
15.6.1 ASP.NET 登录控件概述.....413	17.1.1 什么是 Ajax.....450
15.6.2 登录控件.....414	17.1.2 Ajax 历史.....450
15.6.3 创建用户向导控件.....414	17.2 Ajax 工作原理.....451
15.6.4 密码恢复控件.....415	17.3 微软 ASP.NET Ajax.....453
15.6.5 修改密码控件.....415	17.3.1 ASP.NET Ajax 概述.....453
15.6.6 其他控件.....415	17.3.2 ScriptManager 控件.....453
15.7 ASP.NET 网站管理工具.....416	17.3.3 ScriptManagerProxy 控件.....455
15.7.1 ASP.NET 网站管理工具概述.....416	17.3.4 UpdatePanel 控件.....457
15.7.2 用户管理.....417	17.3.5 UpdateProgress 控件.....458
15.7.3 角色管理.....418	17.3.6 Timer 控件.....459
15.7.4 访问规则管理.....420	17.3.7 ASP.NET Ajax 与 Web Service 的配合.....460
15.7.5 其他配置.....421	17.3.8 ASP.NET Ajax 应用示例: 无 刷新留言板.....463
15.8 成员资格和角色管理 API 编程.....422	17.4 ASP.NET Ajax Control Toolkit.....465
15.8.1 成员资格 API 概述.....422	17.5 Ajax 的优势与不足.....466
15.8.2 角色管理 API 概述.....423	17.5.1 Ajax 应用优势.....466
15.8.3 创建、修改、获取和 删除用户.....423	17.5.2 Ajax 的不足.....466
15.8.4 验证和解锁用户.....425	17.6 脚本调试.....467
15.8.5 角色管理.....426	17.7 回顾与总结.....467
15.8.6 角色验证.....428	第 18 章 性能与安全.....468
15.8.7 其他重要成员.....428	18.1 性能优化概述.....468
15.9 回顾与总结.....429	18.2 编码级别性能优化.....468
第 16 章 Web Part 让网站个性化.....430	18.2.1 数据访问.....468
16.1 Web Part 概述.....430	18.2.2 字符串操作.....469
16.2 进一步了解 Web Part.....431	18.2.3 释放资源.....470
16.2.1 Web Part 的层次结构.....431	18.2.4 其他优化.....471
16.2.2 ASP.NET 中的 Web Part 控件.....431	18.3 页面级别性能优化.....472
16.3 使用 Web Part.....432	18.3.1 页面缓存.....472
16.3.1 创建一个简单的个性化页面.....433	18.3.2 视图状态.....473
16.3.2 页面显示模式介绍.....434	18.3.3 页面加速.....473
16.3.3 在页面中添加 Web Part.....437	18.3.4 其他优化.....473
16.3.4 在页面中编辑 Web Part.....439	18.4 构架级别性能优化.....474
16.3.5 Web Part 的导入和导出.....442	18.4.1 数据缓存.....474
16.3.6 Web Part 之间的通信.....444	18.4.2 分布式程序.....474
16.4 回顾与总结.....449	18.4.3 分布静态资源.....475

18.4.4 其他优化	475	第 20 章 打包与部署	497
18.5 配置与部署性能优化	475	20.1 发布网站	497
18.5.1 网站配置	475	20.2 打包网站	500
18.5.2 网站部署	476	20.2.1 创建 Web 安装项目	500
18.6 安全概述	476	20.2.2 Web 安装项目的细节	502
18.7 页面与编码相关安全防护	477	20.3 回顾与总结	506
18.7.1 控件与页面	477	第 21 章 学成出师——单点登录系统	507
18.7.2 状态机制	478	21.1 单点登录的例子	507
18.7.3 SQL 注入	478	21.1.1 单点登录概述	507
18.7.4 提示信息	479	21.1.2 单点登录基本思想	507
18.8 构架相关安全防护	479	21.2 实现单点登录	508
18.8.1 隔离数据库	479	21.3 使用令牌增进安全	516
18.8.2 独立的操作日志模块	479	21.4 总结和概述	525
18.9 服务器配置相关安全防护	480	第 IV 部分 扩展篇	
18.9.1 细化数据库权限	480	第 22 章 常见的数据访问方式	528
18.9.2 SSL/HTTPS	480	22.1 数据访问概述	528
18.10 其他安全措施	480	22.1.1 常见的数据访问方式	528
18.10.1 防恶意提交	480	22.1.2 回顾 ADO.NET 直接方式	528
18.10.2 跳转令牌	480	22.1.3 回顾强类型 DataSet 方式	529
18.10.3 重复确认身份	481	22.2 ADO.NET 封装方式	530
18.10.4 防跨网站脚本	481	22.3 从数据访问中解放——ORM 方式	531
18.11 回顾与总结	481	22.3.1 ORM 概述	531
第 19 章 调试与错误处理	482	22.3.2 使用 NHibernate 制作留言簿	531
19.1 编码习惯	482	22.4 ORM 的妥协——Sql Map 方式	542
19.1.1 代码格式化	482	22.4.1 Sql Map 概述	542
19.1.2 命名	483	22.4.2 使用 IBatisNet 制作留言簿	542
19.1.3 其他	484	22.5 随心所欲——代码生成方式	549
19.1.4 代码注释	484	22.5.1 代码生成概述	549
19.1.5 版本跟踪	486	22.5.2 代码生成方案举例	549
19.2 发现错误	486	22.6 回顾与总结	557
19.2.1 编译错误	486	第 23 章 分布式应用程序	559
19.2.2 断点调试	487	23.1 分布式应用程序概述	559
19.2.3 跟踪应用程序	491	23.2 Web 服务	560
19.3 处理错误	492	23.2.1 创建 Web 服务	560
19.3.1 异常概述	492	23.2.2 HTTP-GET 方式的 Web 服务	564
19.3.2 捕获和处理异常	493		
19.3.3 自定义错误页	495		
19.4 回顾与总结	496		

目 录

23.2.3 异步调用 Web 服务	565	24.2 多层构架	583
23.3 .NET Remoting	567	24.2.1 多层构架概述	583
23.3.1 第一个 .NET Remoting 应用程序	567	24.2.2 使用三层构架重写留言板	585
23.3.2 Remoting 的信道	571	24.3 依赖注入框架	592
23.3.3 使用配置文件增加灵活性	573	24.3.1 依赖注入概述	592
23.3.4 使用接口降低耦合	575	24.3.2 使用 Spring.NET 进行 依赖注入	596
23.3.5 使用 Windows 服务承载 远程对象	578	24.4 回顾与总结	598
23.3.6 异步操作	581	第 25 章 学习资源	599
23.4 回顾与总结	582	25.1 官方资料	599
第 24 章 分层与构架	583	25.2 技术博客	599
24.1 分层与构架概述	583	25.3 技术网站	600
		25.4 技术论坛	600

第 I 部分 基础篇

或许不是每一本入门的书都会讲基础，或许你也不喜欢看枯燥的基础，但是笔者认为对于任何一门计算机技术来说，打好基础是至关重要的。如果你没有任何 Web 开发的经验，还是建议你在理解了这些基础知识后再开始尝试编程。

我们知道，计算机技术（无论是软件还是硬件）的发展是很快的，软件一年出一新版本，硬件一年出一新代号，对于这样一个高速发展的行业，很多年龄偏大的计算机爱好者往往会对编程望而却步。但是仔细分析可以发现，软件硬件所谓的一年一新往往不是本质上的变化。举一个 Web 开发的例子来说，微软的 Web 开发技术从 ASP 到 ASP.NET 1.1 一直到现在的 ASP.NET 2.0，都需要用到数据库、网络、客户端脚本的知识。从近期的发展来看，Web 编程的基础知识对于今后再学习任何一门其他 Web 编程技术来说都是适用的。