

大师讲堂

ASP.NET 3.5

开发范例精讲精析

229 个网页范例，专注于解决实际开发问题

[基于C#]

华语世界公认最优秀的程序设计图书作家章立民先生
带领最专业的技术团队，全面剖析ASP.NET 3.5的技术精髓

本书完整范例程序及相关数据库文件
章立民老师亲自录制的187分钟高清视频技术教程
——书本之外的增值部分，同样精彩



章立民（中国台湾） 编著
六度获选微软最有价值专家（MVP）

Microsoft® 微软专家 热点技术书库
Microsoft® 微软技术专家 章立民 作品系列

大师讲堂

ASP.NET 3.5

开发范例精讲精析

[基于C#]

章立民（中国台湾） 编著
六度获选微软最有价值专家（MVP）

内 容 提 要

ASP.NET 是目前 Web 开发的主流技术之一,而 ASP.NET 3.5 是其最新版本。本书全面介绍了 ASP.NET 3.5 的技术精髓,以帮助读者快速学习或者过渡到 ASP.NET 3.5。

书中首先介绍了 .NET Framework 3.5 和 ASP.NET 3.5 的新功能,并对 ASP.NET 网页进行了详尽的介绍,包括代码模型、运行模式、表达式语法、重定向网页、网页间的数据传递等;接下来对 ASP.NET 3.5 的主要控件进行了细致探讨,并用大量的网页范例示范了各控件的技术要点及开发应用技巧,包括验证控件、SqlDataSource、ObjectDataSource、列表控件、GridView、DetailsView、FormView 和 ListView 等。

配书光盘提供了书中范例的源代码及范例所需的数据库文件,并有章立民研究室原声录制的高清晰技术视频。本书可以作为初、中级 ASP.NET 软件开发人员的学习用书,特别适合有一定 ASP.NET、ASP 基础的读者。

图书在版编目(CIP)数据

ASP.NET 3.5 开发范例精讲精析. 基于 C#/章立民编著.
北京: 科学出版社, 2009
(大师讲堂)
ISBN 978-7-03-023836-8
I. A… II. 章… III. ①主页制作—程序设计 ②C 语言—
程序设计 IV. TP393.092 TP312

中国版本图书馆 CIP 数据核字(2008)第 211507 号

责任编辑: 何立兵 / 责任校对: 叶翠芹

责任印制: 科 海 / 封面设计: 林 陶

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>

北京市艺辉印刷有限公司印刷

科学出版社发行 各地新华书店经销

*

2009 年 2 月 第 一 版

开本: 16 开

2009 年 2 月第一次印刷

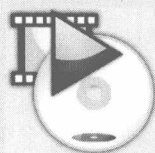
印张: 45.25

印数: 0 001-4 000

字数: 1242 千字

定价: 89.00 元(含 1CD 价格)

(如有印装质量问题, 我社负责调换)



关于光盘

本书配套光盘包括两部分，书中所有网页范例的源代码和“Visual Studio 2008 全新酷炫功能——Web 开发篇”视频文件。

本视频由章立民研究室原声录制，共10讲，收录了Visual Studio 2008的最新功能，为本书之外的附赠内容，精彩超值。全部视频播放时长187分钟，具体内容如下表所示。另外，我们还提供了与视频内容相配套的.sln项目文件。

Visual Studio 2008 全新酷炫功能 —— Web 开发篇

	视频标题	时长	视频目的说明
第1讲	善用 Visual Studio 2008 全新的 Web 设计工具	0:29:51	1. 介绍新的“设计”视图 2. 介绍全新的 CSS 设计工具
第2讲	如何使用全新的 CSS 设计工具快速建立与编辑CSS 样式表	0:15:56	1. 示范如何使用“应用程序样式套用”工具栏建立的样式规则 2. 示范如何在 CSS 编辑器中添加、修改样式规则 3. 示范如何使用“管理样式”窗口来移动或复制样式规则
第3讲	在 Visual Studio 2008 中如何制作嵌套的母版页	0:14:27	1. 示范如何在子母版页套用父母版页 2. 示范如何在页面中套用父母版页与子母版页
第4讲	如何使用全新的 ListView 控件来显示、分页与排序数据	0:19:28	1. 示范如何为 ListView 控件建立数据绑定 2. 示范如何在 ListView 控件中添加分页功能 3. 示范如何让 ListView 控件具备排序功能
第5讲	如何使用全新的 ListView 控件来编辑数据	0:11:18	1. 示范如何为 ListView 控件建立数据绑定 2. 示范如何让 ListView 控件具备编辑功能
第6讲	如何使用与自定义全新的 JavaScript IntelliSense 功能	0:18:37	1. 示范在编写 JavaScript 程序代码时如何使用 IntelliSense 功能 2. 示范如何善用“AJAX 扩充功能”简化 JavaScript 语法 3. 示范如何为 JavaScript 函数添加批注说明
第7讲	如何有效调试 JavaScript	0:17:46	1. 示范在调试 JavaScript 程序代码时，需要启用哪些设定 2. 说明“脚本文件”节点中的文件 3. 示范如何在“实时运算”窗口中，显示或改变某个对象或变量的值
第8讲	如何利用客户端 JavaScript 异步调用 WCF 服务	0:18:27	1. 示范如何添加“已启用 AJAX 的 WCF 服务”项目 2. 示范如何编写 WCF 服务 3. 示范如何使用 JavaScript 调用 WCF 服务
第9讲	如何使用全新的 Silverlight 控件来快速添加Silverlight 功能	0:18:34	1. 说明如何调整 ASP.NET 3.5 Extensions 的设置，以便使用 ASP.NET 3.5 Extensions 的范本 2. 了解如何使用全新的 Silverlight 控件来快速添加 Silverlight 功能
第10讲	如何使用全新的 MediaPlayer 控件来制作 Silverlight 多媒体界面	0:22:00	1. 了解如何使用全新的 MediaPlayer 控件来制作 Silverlight 多媒体界面 2. 说明如何为 MediaPlayer 控件添加多媒体文件并变更播放器的外观 3. 如何使用 JavaScript 控制 MediaPlayer 控件的播放功能

提示：本书视频的讲解基于Visual Basic项目，尽管如此，并不影响作者讲解Visual Studio 2008的超强功能，也不会对您学习本书的内容造成困扰。希望本附赠内容可以进一步拓展您对Visual Studio 2008的认识。

序

是序也不是序

对我而言，写一本书的序言，往往是最痛苦的时刻。就好比跑一场马拉松，最后的50米却是悬崖峭壁，但必须奋力爬过，才算完成与完整。一直以来，我都不希望在序言中对技术高谈阔论，因为这样既展现不出个人风格，也看不出任何人文素养。如果你发现我们某本书的序言，竟在述说技术的“种种”，那表示已经无话可说，只好以此搪塞。但是若要谈心情转折，有时又会流于为赋新词强说愁，不仅显得矫情，更是“肉麻”。这种追求完美的矛盾情绪，最后其实是苦了自己。

为了寻找灵感，今天特地去了台大校园，希望假日校园里的宁静气氛可以让我多点灵感。不过很多事情都是事与愿违，心情没静下来不说，倒是兴致勃勃参加了该校为毕业生举行的活动，不仅到两行垂柳赛睡莲的“醉月湖”划船，还骑一匹高大骏马绕了一小圈，最后更观赏了一场棒球大战。我想即使是“台大人”，有这样经验的也不多吧！我只能说，生命中还是时时充满惊喜的。

其实这半年来发生了太多事情，几乎每一件都是我生命中的第一次，让我的心情犹如过山车一般上下起伏。还好年纪不小，表面上还能故作镇定，工作不仅没受到影响，效率甚至更高。不过相知相识多年的老友，即使多年未见，从国外回来一看到我，还是能从眉宇间察觉出那种说不出的不安。真的，很多老友都在我生命中扮演重要角色，给我很多支持。记得远在对岸奋斗的J，只因某夜突然梦见我，担心我是不是有什么事情，隔天就与我联系，害我当场没感动得痛哭流涕。

人生真的很短暂。许多仿佛是昨日之事，其实都已过数年，甚至10年。我期望自己的未来，少点犹豫，多点行动，带着勇气，努力追寻，持续奋斗。以此共勉！

章立民

2008.5.4

笑谈我的学习之路

读者在博客的留言板上留言，希望我能够分享 .NET 的学习过程、心态以及方法。于是写下本文，与大家分享。

在信息软件界讨生活，要放轻松是谈何容易。说“笑谈”是有点故作轻松了，“心路历程”听起来又有点恶心，讲到“传承”吗，那更是不敢当，总之，大家就是放轻松，当作生活经验的交流。将我近 20 年来的经历与心得，毫不保留但简短地跟大家分享。更重要的是，我希望大家在读完本文后，都能够更清楚以及坚定地选择你未来所要走的路。

技术快速演进的不变定律

纵览人类文明的演进，大抵都建立在“今日会比昨日好、明日要比今日强”的基本史观之上。回顾过往的信息技术洪流，20 世纪 80 年代末还在钻研 DOS 系统、汇编语言，90 年代初却得踏进 Windows 3.1、Windows SDK、DDE 等新架构所延伸的密林，大气还没喘上一口，却又在 90 年代中期碰上了 32 位 Windows、MFC 与 OWL/VCL、OLE2 所构成的技术风暴，而 Web 也在此时开始火热，主从式架构也快速增加为三层式、甚至多层式的 Windows DNA，正当庆幸千禧年的 Windows 架构还在掌控之中，.NET 技术的跃进又像一波波不断来袭的大浪。

身处速度快、变化大的信息技术洪流，的确必须随时准备好面对更快、更好、更强的明日世界。但是，技术的本质是什么？学习的瓶颈该如何突破？如果只是一味盲目乱冲，而不适时让自己缓下来思考，很容易被洪流吞噬。面对技术洪流，你的生存之道是什么？你是否也思考了技术的本质以及学习的方法？更重要的是，你真的准备好面对这永无止尽的磨练（或者说折磨）吗？

你为什么进这一行

正所谓男怕入错行，女怕嫁错郎（我必须说，这观念其实是迂腐的）。所以在我们分享所谓的经验之前，应该先回到源头，问问自己，为什么入这一行。在两岸，常常会有一些年轻的程序员感到彷徨而来求助于我。我通常第一个问题就是问对方，你为什么进这一行。是因为你在学校原本就读信息相关的专业而自然进了这一行，还是当时自己或家人觉得电子与信息相关产业非常热门，因此进了这一行。如果你是因为觉得产业热门才进了这一行，而且是在 1999 年之后才入行的，那我只能送你一句话，那就是“夕阳无限好，只是近黄昏”。所以，如果你在午夜梦回的时候常常会对自己的工作有所质疑，那真的应该好好思考思考。

我所要强调的是,你对信息软件真的有兴趣吗?你真的喜欢这样的工作吗?你顶得住压力吗?更重要的是,你快乐吗?能够把一件事情做好,并不表示你真的喜欢,也不表示你真的快乐,这只代表你有足够的能力。如果这真的是你的兴趣,你真的热爱,就算辛苦,也会乐在其中。

身处这一行,工作日与假日加班是常态。我常常看到许多年轻人,不过就是假日加个班,就觉得自己倍感辛苦,备受委屈。如果你也常有这种心态,赶快转行吧!更可怕的是,自己辛苦也就算了,家人与亲密爱人常常无法理解与谅解,同样地,他们的抱怨也成了常态。干这行,还有许多后遗症,比如身体变形,秃头,满头白发,心情很难放松,等等。

我谈这些,并不是要你离开这个行业,我的目的是希望大家再次肯定自己是喜欢这个行业的,不论你是开发人员或者 IT Pro,都能够更加清楚自己的选择,然后坚定地走下去。

技术路线的选择

在你确定自己适合走技术这条路,也决定要走技术这条路之后,接下来面临的的就是技术路线的选择,而这通常也是学生或初入行者最大的抉择。为什么呢?因为这关系到他未来的就业,他希望自己押对宝。然而为什么会难以抉择呢?因为目前看不出何者拥有绝对的优势。据说是 .NET 小赢。但是我们必须了解,任何一种技术架构都有其特性与优劣所在,然而难就难在 .NET 与 J2EE 都非常庞大,我们很难两者都精通。因此我说成功没有一定的规则可言,失败却有,那就是“两面讨好者”。

我常常看到有许多人喜欢比较 .NET 与 J2EE。说句实在话,这两种技术路线都有极深的内涵,单单看技术架构,稍微用点其中的技术进行开发,就足以评断两者的优劣吗?这未免太自以为是,大放厥词。我投注了极大的心力研究 .NET,才认为能够稍微随心所欲地掌握其内涵,也才能够真正体会其中的优点与缺点。我相信 J2EE 也是一样的道理。问题是,我时间有限,精力有限,能力有限,不可能同时研究这两者,既然如此,何德何能去评论呢?

我要说的是,评论诸项技术时,最忌讳的就是“成见”以及那股莫名的“情绪”,或者说“情结”。当你有了这样的心态,如何以客观的心情去评量各种技术呢?奉劝大家,在选择技术的时候,不需要有太多的情绪与爱恨情愁。我知道有很多人就是看微软不爽,看微软赚那么多钱就觉得很讨厌,因为我讨厌微软,所以我不用微软的技术,所以我选择 J2EE、Java、Oracle。微软的技术真的不好吗?同样地,你也不需要因为非常喜欢微软,所以就讨厌 J2EE、Java、Oracle。身为一位技术人员,一定要抛开这些成见与情绪,因为一旦有这些成见,就会影响你的判断力,也因此很可能与好的技术失之交臂。所以我的态度是,不论技术是谁推出的,一定要用开放与正面的态度去看待。正所谓“知己知彼,百战百胜”,不论是自我学习,还是在市场上竞争,切忌闭门造车。

其实只要你真正投入,掌握技术的精要,自然能彻底发挥其特性,进而开发出优质的应用程序。如果你学东西总是半吊子,那即使你用的是伟大的技术,写出来的东西也是平淡无奇,甚至上不了台面。

我选择技术的原则非常简单，第一看技术的进入障碍；第二看技术有没有未来性。所谓进入障碍，当然就是跟我目前的技术是不是有延续性，是不是能够有良好的衔接，这是为了确保我过去投入的心血不会轻易付诸流水。就未来性而言，就是看技术有没有前景，说得露骨点，应该说有没有“钱”景，毕竟技术必须叫好，也要叫座，才是好技术，那种孤芳自赏者，通常没有好下场。没错，这两个原则有时候是彼此相冲突的。放眼技术的演进史，有时候会因为某种革命性架构与装置的推出，导致引发大跃进式的演进。在这种时候，未来性的考虑当然更胜于进入障碍。

其实在许多时候，技术（或者说技术路线）的选择也没有那么难，因为市场可能已经帮你做了决定。如果真的无法自行判断，就跟着潮流走吧！

不需要跟技术“谈恋爱”

别让“执着”害了你。就好比我们劝人在投资理财的时候不要跟股票“谈恋爱”一样，你也不需要跟技术“谈恋爱”。没错，你当初看好某一家公司的前景，买了该公司的股票准备长期投资，可是当你发现公司营运状况不如预期或者出现更大的危机时，难道还要抱着股票不放吗？你希望你的股票变成废纸吗？没错，你现在抛售股票已经受伤，已经赔钱，可是如果不当机立断、壮士断腕，损失将会更惨重。投资理财如此，技术的选择也是如此。

拿得起，也要放得下。当发现所选择的技术已经失去优势，应该毫不犹豫地转换。没错，你已经投入了许多的时间和精力学习该项技术，要放弃，谈何容易。不过 IT 这个行业就是如此，过去被奉为圣经、归镍的技术架构可能在短短的时间就被淘汰出局。所以我们说，你也必须保持一个开放的态度，随时注意趋势的变化，不要恃才而骄，自以为了不起。就好像戏水溺毙的人多半是游泳高手一样，因为他自认功夫过人一等，不会有任何危险，完全没有任何的警戒心，因此反而比别人死得更快。

我为什么要特别提到“不要跟技术‘谈恋爱’”，因为我之所以会在 1999 年急转直下，陷入低潮，就是犯了这个错误。在很久之前，我靠 FoxPro 成名，我的存款、房子、车子都是靠 FoxPro 这只“狐狸”赚来的。可是当时已经有一些明显的迹象显示 FoxPro 未来将不会是微软的发展重点。这些迹象包括：FoxPro 在美日两大市场卖得非常不好，Tech-ED 的 FoxPro 场次人数递减等。甚至连微软内部的高层人员都要我小心，可是我却不信邪。最后的结果就是，承受痛苦。

我是如何学习的

正所谓万丈高楼平地起，要把功夫练好，马步一定要蹲得稳。我想很多人最好奇的，就是我如何学习新技术。说真的，我用的都是笨方法。学习技术，一定要先把架构与原理彻底了解，然后才能晋升到技巧的活用。而这中间的窍门，就是不停且大量的实践，并将结论有组织地整理起来。

每当我阅读一段原理与说明时，都会尽可能实际动手操作一下或编写程序验证，一方面可以确认自己的理解是正确的，一方面也可以更加熟悉并体会其内涵，了解其应用层面。当然，人的记忆力是有限的，最终的验证结果与结论必须要进行系统地整理，以方便未来能够快速查阅。

这中间的过程不仅枯燥，更是乏味，而且很寂寞，然而却是必经的过程。就我的了解，业界的开发人员，可能最缺乏的就是这一段学习过程，毕竟大家都急于对老板与客户负责，交差结题，因此常常会流于追逐实用的技巧。但是大家别忘了，彻底了解技术架构，可能会比追逐新工具来得更长久，

毕竟许多原理都是相通的。就 .NET 而言, 从 2002 年推出到目前的 .NET 3.5, 基本的大架构并没有什么不同, 只是一直在既有的基础上累加并扩展, 如果你当初就能够秉持正确的学习原则, 或许现在会感觉更加如鱼得水。

如何让自己与众不同

如果要在这一产业长久发展, 你必须让自己与众不同, 而不同之处不仅仅是技术。

我很喜欢问开发人员一个问题: “除了高人一等的技术能力之外, 你还有什么?” 我常跟别人说, 你再会写程序, 你也只是个程序员。这句话听起来或许很伤人。不过就像过去很多人问我写书要写到几岁一样, 我以前从来不觉得这是个问题, 但是当年纪大了, 我开始思考这个问题。因此我现在问你, 你要写程序写到几岁呢? 难道你要一辈子写程序吗? 你以为每一位会技术的人都可以像 Bill Gates 一样吗? 你以为 Bill Gates 就只是会技术吗? 他没有其他的特质吗?

也不是说你不能一辈子写程序, 不能一辈子当技术人员。因为有些人真的不适合当主管, 不适合开公司做生意。如果真的想要长久地做技术研究, 你就需要一个比较稳定的环境。可能是在学校或研究单位, 或者大型的私人企业。大家要早点想这个问题, 毕竟中年转业不容易啊。

我也常告诫人, 不要让高人一等的技术能力成为反动力。正所谓“工作的金字塔, 一人抵三人。”你技术能力强, 以一敌三, 老板为什么要给你升职, 因为要用三个人补你的缺, 而且你看起来真的不像个主管。为什么升任主管的常常是你认为技术不太行的, 只会“耍嘴皮子”的。结果是, 你花了很长的时间与很大的努力来证明自己, 可是等你年纪大了, 公司可能觉得你太老, 不想给你机会。唉! 真是残酷啊!

或许你可以思考如何进一步融入“热情”、“人文素养”与非技术的“专业素养”, 或许这样可以让你更发光发热。我在 2007 年之前的好几年都不讲研讨会, 因为我不觉得我可以讲得比以前更好, 我觉得难以再超越过去的成绩, 我有疲惫感, 也就是说我没有热情。与其勉强上台, 还不如不上台。假设我自己都不热爱这样的工作, 如何让听众去热爱呢。不要忘记, 任何人比你都敏感。千万不要以为可以浑水摸鱼, 蒙混过关。

我还想问你, 你的特质是什么。为什么同一首歌, 由不同的人演唱, 会有完全不同的效果。因为他捉住了其中的感觉, 能做最佳的诠释。反过来说, 量身订制, 因此你应该寻找最适合你的表现方式。

千万记得, 在演讲、讲课或做简报时, 是不可以忘我的, 要随时与台下观众保持互动。所谓的互动不见得是言语的互动, 而是通过你的眼神、话语和肢体语言来传达信息, 而从听众的表现、坐姿等, 都可以看出他们的反应, 并随时动态调整。因此我较不喜欢列出太细的项目, 保持动态调整的弹性, 而且准备的内容应该要比实际的会议时间来得长和丰富。

最后建议科技人应当走出技术的象牙塔, 培养良好的人际关系, 展现不凡的热情与活力。

抓住机会，秀出自己

技术人员一般来说都比较内向，比较沉默，不喜欢表现。可是我真的鼓励大家，抓住机会，秀出自己（Present yourself）。当初我第一次到微软并且帮微软讲研讨会，根本就是毛遂自荐的。重点是，当别人给我机会时，我充分抓住了。

成功是性格、努力与机会的结合。我们常说性格决定命运，努力要有方法和策略，最后还需要天时、地利以及人和。你说路遥知马力，日久见人心，问题是有机会表现时，你闷不吭声，好像不存在。其结果是，人家还以为你能力不够。

当然，如果你是金玉其外，败絮其中，也撑不了太久。但是如果你真的有能力，给你一个良心的建议：“不要只做沉默的工蜂”。毕竟机会稍纵即逝，逝者不可追。

在这里给大家提供一组真实的统计数据，当你说话的时候，人们接收到的信息是什么？请注意，有 93% 来自肢体语言和语调。

- 55%：外型与肢体语言。
- 38%：语调。
- 7%：话语（内容）。

为什么我要特别强调上述数据，因为我听很多技术人员讲话的时候，真的很想扁人。不是说话的音调没有高低起伏，就是喃喃自语不知所云，不然就是自顾自地讲，完全沉醉在自己的话语和情绪中。唉！真是要昏倒。

另外还有一组重要的数据提供给大家。一场演讲后，人们还记得什么：

- 过了 1 小时，人们会忘掉 50% 的内容。
- 过了 1 天，人们会忘掉 80% 的内容。
- 过了 1 周，人们会忘掉 95% 的内容，剩下的记忆是：故事、例子、亲身经历。

这也就是为什么我们即使谈的是技术性的研讨会，也要举许多的实例与成功案例，因为这样印象才会深刻。

反省的能力

策略一定要动态调整。人们往往会活在过去的成功模式中，然而今非昔比，策略应时时调整。

千金难买早知道，千金难买回头望。也许我们无法抓住机会，但是藉由不停地反省与修正，也能让自己朝对的方向前进。

健康是最大的财富

大家都听过这句话吧：“健康的身体是 1，没有这个 1，后面的 0 将毫无意义”。美国前总统里根曾经说过：“It is true hard work never killed anyone, but I figure, why take the chance?”（努力工作确实不会致人死命，但我想，何必冒险？）

当然，我们之前一直提到要努力，加班也不要抱怨，问题是没人要你熬夜与晚起，你就不能早睡早起吗？时间一样长得很，更重要的是，这样对身体有益。很多东西在失去了之后还可以找回，但是健康不能。刻骨铭心的恋情可能会永远占驻你心中的角落，不时还会隐隐作痛，问题是它不会夺去你的生命；错过的电影，还有下一场，健康可能没有；这个男生不爱你，没关系，下一个男人会更好，只要你有健康的身体。

过去六七年来，我一直保持运动的习惯，饮食也是清淡得不行，有人说我过得是清教徒的生活。运动不仅让身体更舒畅，也让大脑能够休息与清醒，为下一次的吸收做好准备。建议大家也尝试看看吧！

结论

人在好的时候要想到不好的时候，人在不好的时候不要去想好的时候，而是应该静下心来思考方法与策略，然后迅速脱离谷底。当然，我自己对未来也有很多期许，希望能再创高峰。

有人说：“发现一瓶好水，令人心旷神怡；发现一位好作者，令人眉开眼笑。”这么多年来，我写书一向注重质量，期许自己能够成为让人眉开眼笑的好作者。过去如此，现在如此，未来也将如此。近 20 年来，经过起起伏伏，风风雨雨，今天还站在这里，我是心存感激的。下次再见，谢谢大家。

章立民

2008.6.23

目录

Chapter 01 开门见山：从实战开始学起	1
1-1 实际创建一个 Web 项目与数据库访问网页	2
1-2 ASP.NET 3.5 的主要组成架构	13
1-2-1 网页与控件架构	13
1-2-2 ASP.NET 编译器	14
1-2-3 安全性基础架构	14
1-2-4 完善的状态管理功能	14
1-2-5 应用程序配置设置	14
1-2-6 健全状况监视与性能机制	15
1-2-7 调试机制	15
1-2-8 XML Web 服务架构	15
1-2-9 可扩展的主机服务与应用程序生命周期管理	15
1-2-10 可扩展的设计工具环境	15
1-3 ASP.NET 3.5 新功能概述	15
1-3-1 全新的 ListView 控件	16
1-3-2 全新的 DataPager 控件	19
1-3-3 全新的 LinqDataSource 控件	19
1-3-4 全方位的 AJAX 开发能力	24
1-3-5 超好用的 JavaScript IntelliSense 功能	25
1-3-6 ASP.NET 合并工具	26
1-3-7 完美整合 IIS 7.0	26
1-3-8 全新的 CSS 设计工具	26
1-3-9 功能更完善的 Web 应用程序项目	27
1-3-10 多重目标架构的 Web 应用程序	28
1-3-11 创建与使用 WCF 服务	28
1-4 什么是 .NET Framework 3.5	29
1-5 ASP.NET 3.5 平台的系统需求	29
结语	30

Chapter 02	ASP.NET 应用程序的灵魂: ASP.NET 网页	31
2-1	实现一个 ASP.NET 网页	32
2-2	ASP.NET 网页的功能特性	35
2-3	ASP.NET 网页能够完成哪些操作	36
2-4	ASP.NET 网页的代码模型	37
2-4-1	单一文件网页模型	38
2-4-2	代码隐藏网页模型	39
2-4-3	该选用哪一种网页模型	41
2-4-4	编译与部署	41
2-5	探讨 ASP.NET 的 Page 类	41
2-5-1	单一文件网页	42
2-5-2	代码隐藏网页	43
2-6	ASP.NET 网页的运行模式	43
2-6-1	往返	43
2-6-2	重建网页	45
2-6-3	基于事件的模型与线性处理模型	45
2-7	ASP.NET 网页的生命周期阶段	46
2-8	ASP.NET 网页的生命周期事件	47
	网页范例1 示范如何利用PreInit事件动态设置网页的主题	48
	网页范例2 示范如何使用网页事件Load与PreRender	50
2-9	ASP.NET 网页生命周期的其他考虑	53
	网页范例3 示范如何动态设置网页的区域信息	53
2-10	ASP.NET 服务器控件事件模型	55
2-10-1	服务器控件与网页的事件集	56
	网页范例4 示范如何使用CheckBox控件的CheckedChanged事件	57
2-10-2	事件参数	59
	网页范例5 示范如何使用ImageClickEventArgs类型参数	59
2-10-3	服务器控件中的回发与非回发事件	60
	网页范例6 示范如何为网页动态应用样式表文件	60
2-10-4	冒泡事件	63
2-10-5	事件委托与 AutoEventWireup 属性	63
2-10-6	应用程序事件与会话事件	63
2-11	如何创建事件处理例程	64
2-12	如何在运行时绑定事件处理例程	66
2-13	如何将多个事件绑定到同一个事件处理例程	66
2-14	如何判断是哪一个服务器控件触发了事件	67
	网页范例7 示范如何判断触发事件的服务器控件	67
2-15	服务器控件如何响应客户端事件	69

2-15-1 活用 OnClientClick 属性	69
网页范例8 示范如何使用OnClientClick属性来打开网站	70
网页范例9 示范如何使用OnClientClick属性进行删除确认	71
网页范例10 示范如何在GridView控件中进行删除确认	72
2-15-2 客户端属性	73
网页范例11 通过客户端属性来制作鼠标移入与移出效果	74
网页范例12 示范如何动态创建客户端属性来处理客户端事件	75
结语	76

Chapter 03 万丈高楼平地起：ASP.NET 网页的重要语法和技巧

3-1 怎样用好 Web 应用程序根运算符 ~	78
3-2 程序代码块语法	78
网页范例1 示范怎样使用程序代码块语法	79
3-3 服务器端批注语法	80
3-4 服务器端#Include 指示符语法	81
网页范例2 示范怎样使用 #Include来插入页眉和页脚	81
网页范例3 示范怎样使用包含文件来提高设计效率	82
3-5 表达式语法 <%\$... %>	83
3-5-1 使用<%\$... %>取得连接字符串	83
网页范例4 示范使用<%\$... %>取得连接字符串	84
3-5-2 使用<%\$... %>取得应用程序设置	85
网页范例5 示范使用<%\$... %>取得应用程序设置	85
3-5-3 使用<%\$... %>取得资源文件设置	86
网页范例6 示范使用 <%\$... %> 取得资源文件设置	86
3-6 怎样在网页中检测浏览器的类型和版本	89
网页范例7 示范如何检测浏览器或客户端设备的相关信息	89
网页范例8 根据浏览器的类型和版本为网页应用不同的CSS样式表文件	90
3-7 重定向网页方法一：超级链接	92
3-8 重定向网页方法二：Response.Redirect 方法	93
网页范例9 示范怎样根据用户计算机的区域语言设置来重定向网页	93
3-9 重定向网页方法三：Server.Transfer 方法	94
3-10 重定向网页方法四：跨网页传送——PostBackUrl 属性	95
3-11 重定向网页方法比较表	96
3-12 网页之间怎样使用查询字符串传递数据	97
网页范例10 示范如何使用查询字符串在各网页之间传递数据	98
3-13 Server.Transfer 方法的数据传递方式	99
网页范例11 示范使用Server.Transfer方法时，怎样在目标网页上读取源网页的控件和公有属性的值	100
3-14 跨网页传送的数据传递方式	103

3-14-1 源网页和目标网页位于相同的 Web 应用程序中	103
网页范例12 使用PreviousPage属性和@ PreviousPageType指示符来读取源网页的控件和公有属性值	104
网页范例13 使用PreviousPage属性和@ Reference指示符来读取源网页的控件和公有属性值	110
3-14-2 源网页和目标网页位于不同的 Web 应用程序中	111
网页范例14 示范不同Web应用程序中网页的跨网页传送	112
3-15 判断网页是怎样被调用的	113
网页范例15 示范如何判断网页是怎样被调用的	115
3-16 怎样设置网页的默认焦点控件和默认按钮	116
网页范例16 示范使用默认焦点控件和默认按钮创建快速搜索界面	117
3-17 认识网页指示符	119
3-18 让网页倒数几秒后自动重定向到其他网页	120
网页范例17 示范倒数3秒后自动重定向到特定网页	120
网页范例18 示范如何以程序控制方式为网页添加meta标记	121
结语	122

Chapter 04 界面设计前传: ASP.NET 服务器控件的基本概念 123

4-1 ASP.NET 服务器控件的类型	124
4-2 使用建议	130
4-3 浏览器功能所影响的方面	131
4-3-1 ASP.NET 怎样自动检测浏览器类型	131
4-3-2 改写浏览器类型检测	131
4-3-3 关于 AJAX 功能	132
4-3-4 客户端脚本	132
4-3-5 使用早期版本的浏览器	132
4-4 如何将服务器控件添加到 ASP.NET 网页	134
4-5 以程序控制方式将服务器控件添加到 ASP.NET 网页	136
网页范例1 示范如何动态添加服务器控件	136
4-6 在设计阶段设置服务器控件的属性	138
4-7 以程序控制方式设置服务器控件的属性	138
4-7-1 动态使用常数值设置属性	138
网页范例2 示范如何使用常数值动态设置属性	139
4-7-2 动态使用枚举类型设置属性	139
网页范例3 示范如何使用枚举类型动态设置属性	139
4-7-3 动态设置单位属性	139
网页范例4 动态设置服务器控件的大小和单位	140
4-7-4 动态设置颜色属性	142
4-7-5 动态设置集合中的控件属性	143
网页范例5 示范怎样为ListBox服务器控件动态添加选项	143

4-7-6 使用 Style 对象动态应用样式	144
网页范例6 使用Style对象一次动态应用多种样式到多个服务器控件	144
网页范例7 根据用户所选择的样式来构建Style对象并应用到服务器控件	146
4-8 以程序控制方式设置 HTML 服务器控件的属性	146
网页范例8 示范如何使用Attributes集合取得HTML服务器控件的属性设置	146
4-9 服务器控件和样式表	147
4-9-1 解释外观属性	147
4-9-2 控件样式对象	148
4-9-3 样式对象的优先级和继承	148
4-9-4 直接控制 CSS 样式和类型	149
网页范例9 示范如何使用CssClass属性为服务器控件应用样式类	149
4-9-5 怎样设置样式属性	151
网页范例10 示范如何使用CopyFrom和MergeWith方法来应用样式	153
结语	154

Chapter 05 网页的基本元素：探讨基础的 ASP.NET 服务器控件

5-1 Label 服务器控件	156
网页范例1 将用户所提供的数据编码后赋给Label服务器控件	156
5-2 Literal 服务器控件	157
网页范例2 将用户所提供的数据编码后赋给Literal服务器控件	158
5-3 TextBox 服务器控件	159
5-3-1 访问数据值	159
5-3-2 善用 TextMode 属性	160
5-3-3 限制输入的字符数	160
5-3-4 是否为只读	160
5-3-5 是否自动回发	161
5-3-6 TextChanged 事件	161
5-3-7 设置快捷键	161
网页范例3 结合文本框和数据绑定技巧的快速查询界面	161
网页范例4 TextChanged事件使用示范	162
5-4 CheckBox 服务器控件	163
网页范例5 示范怎样使用复选框的InputAttributes和LabelAttributes属性	164
5-5 RadioButton 服务器控件	168
网页范例6 示范如何使用RadioButton控件制作单选界面	168
5-6 Button 服务器控件	170
网页范例7 示范如何使用Button服务器控件来触发查询操作	171
网页范例8 示范怎样使用命令按钮	172
网页范例9 示范怎样使用命令按钮来动态设置SqlDataSource控件的查询语句	174
网页范例10 示范如何为按钮添加鼠标移入和移出的变化效果	175

5-7 ImageButton 服务器控件	175
网页范例11 ImageButton服务器控件坐标检测示范	176
网页范例12 鼠标指针移到ImageButton上方时的样式变换	177
5-8 LinkButton 服务器控件	177
5-9 HyperLink 服务器控件	177
网页范例13 示范通过HyperLink控件动态创建项目中的网页列表	178
5-10 Image 服务器控件	180
网页范例14 结合HyperLink和Image服务器控件制作一个简单的相册	180
5-11 Calendar 服务器控件	183
网页范例15 制作一个简单的日历界面	183
5-11-1 自定义日历的样式外观	184
5-11-2 自定义日期选择模式	186
网页范例16 Calendar服务器控件的日期选择示范	187
网页范例17 示范如何以程序控制方式选择多个日期	190
网页范例18 示范使用SelectRange方法选择指定的日期范围	191
5-11-3 控制月份导航	192
网页范例19 示范怎样快速导航到某年某月	193
5-11-4 自定义个别日期——重要!!	194
网页范例20 示范如何将自定义内容添加到日期中	195
5-12 ImageMap 服务器控件	199
网页范例21 示范制作一个具有矩形热点的图片地图	202
网页范例22 示范制作一个具有圆形热点的图片地图	203
网页范例23 示范制作一个具有多边形热点的图片地图	204
网页范例24 示范制作一个会触发网页回发的图片地图	205
5-13 FileUpload 服务器控件	208
结语	208

Chapter 06 数据检验的利器：验证控件 209

6-1 验证操作的基本逻辑和思想	210
6-2 验证控件的类型	210
6-3 客户端验证	211
6-3-1 差异性	211
6-3-2 客户端验证对象模型	212
6-3-3 提交客户端验证错误	212
6-4 特殊案例的验证结果	212
6-4-1 空白字段	212
6-4-2 检查其他控件所保存的数据	213
6-5 输入数据非空验证	213
网页范例1 示范如何使用RequiredFieldValidator验证控件及其InitialValue属性——重要!!	215