

迅腾国际“5+N”信息化紧缺人才培养计划系列丛书

ASP.NET高级特性

天津滨海迅腾科技集团有限公司 编著



天津出版传媒集团



天津科技翻译出版有限公司

迅腾国际“5+N”信息化紧缺人才培养计划系列丛书

ASP.NET高级特性

天津滨海迅腾科技集团有限公司 编著

天津出版传媒集团

天津科技翻译出版有限公司

图书在版编目(CIP)数据

ASP.NET高级特性 / 天津滨海迅腾科技集团有限公司编著. — 天津: 天津科技翻译出版有限公司, 2013.1

(迅腾国际“5+N”信息化紧缺人才培养计划系列丛书)

ISBN 978-7-5433-3163-1

I. ① A… II. ①天… III. ① 网页制作工具—程序设计 IV. ①TP393.092

中国版本图书馆CIP数据核字(2012)第307801号

主 编: 邵荣强

副主编: 李国燕 常秀岩

出 版: 天津科技翻译出版有限公司

出 版 人: 刘 庆

地 址: 天津市南开区白堤路 244 号

邮政编码: 300192

电 话: 022-87894896

传 真: 022-87895650

网 址: www.tsttpc.com

印 刷: 廊坊市圣轩印刷有限公司

发 行: 全国新华书店

版本记录: 889×1194 16 开本 15 印张 300千字

2013年 1月第 1 版 2013年 1月第 1次印刷

定价: 98.00元

(如发现印装问题, 可与出版社调换)

目 录

理论部分

第一章 母版页 MasterPage	3
1.1 MasterPage 基础.....	4
1.2 编写 MasterPage.....	5
1.2.1 给应用程序添加 MasterPage.....	5
1.2.2 编写 MasterPage 上的公用代码.....	6
1.3 编写内容页面	8
1.4 指定要使用的 MasterPage	11
1.4.1 在页面级指定使用母版页	11
1.4.2 在 web.config 中指定使用母版页.....	11
1.4.3 指定特定文件夹中的内容页面所使用的母版页.....	11
1.5 使用页面标题.....	12
1.6 访问 MasterPage 中的控件和属性	12
1.7 指定内容区域的默认内容	13
1.8 动态使用母板文件	14
1.9 事件的触发顺序	14
第二章 站点导航.....	16
2.1 基于 XML 的站点地图	17
2.2 SiteMapPath 服务器控件.....	18
2.3 TreeView 服务器控件.....	20
2.3.1 声明式添加 TreeView 节点.....	20
2.3.2 绑定到站点地图和 XML 文件.....	24
2.3.3 定制 TreeView 控件	25
2.3.4 编程使用 TreeView 控件.....	29
2.4 Menu 服务器控件.....	31
2.4.1 声明式添加 Menu 条目	31
2.4.2 绑定到站点地图和 XML 文件.....	32
2.4.3 定制 Menu 控件	33
第三章 主题.....	37
3.1 创建主题.....	38
3.1.1 主题的构成	40



3.1.2 主题中的 CSS	40
3.1.3 主题中的图像	41
3.1.4 主题中的 Skin	41
3.2 主题的应用	44
3.2.1 给单个页面应用主题	44
3.2.2 给整个应用程序应用主题	44
3.2.3 指定某些控件使用特定的外观设置	44
3.2.4 使用母版页的时候应用主题	45
3.2.5 不想受主题控制怎么办?	45
3.2.6 一些常见的主题应用问题	46
第四章 安全性与程序调试	48
4.1 身份验证	49
4.1.1 身份验证的方式	49
4.1.2 基于 Windows 的身份验证	49
4.1.3 基于窗体身份验证	52
4.2 验证特定的文件和文件夹(URL 授权)	56
4.3 VS 2005 在设计期间提供的支持	57
4.3.1 语法通知	57
4.3.2 即时窗口和命令窗口(Immediate and Command Window)	58
4.3.3 任务列表(Tasklist)	58
4.4 跟踪	59
4.4.1 System.Diagnostics.Trace 类和 ASP.NET 的 Page.Trace 属性	59
4.5 调试	62
4.6 异常和错误处理	63
第五章 成员和角色管理	65
5.1 网站管理工具	66
5.1.1 启动管理工具	66
5.1.2 用户管理	67
5.1.3 用户角色	68
5.1.4 访问规则管理	69
5.1.5 应用程序配置	71
5.2 ASP.NET 2.0 的身份验证	72
5.2.1 为成员建立 Web 站点	72
5.2.2 添加用户	74
5.2.3 请求凭证	81
5.2.4 处理验证用户	84

5.2.5 处理密码	85
5.2.6 使用 Membership API	88
5.3 ASP.NET 2.0 的授权	89
5.3.1 使用 LoginView 服务器控件	89
5.3.2 为角色管理建立 Web 站点	90
5.3.3 添加和检索应用程序角色	92
5.3.4 删除角色	95
5.3.5 给角色添加用户	95
5.3.6 从角色中删除用户	95
5.3.7 Roles API 的公共方法	96
5.3.8 角色的存储方式	96
第六章 模块和处理程序	99
6.1 IIS 可扩展性 API 概述	100
6.2 使用 HTTP 处理程序	104
6.3 适用 HTTP 模块	110
第七章 国际化与本地化	116
7.1 文化和地区	117
7.1.1 文化类型	118
7.1.2 线程上的文化设置	119
7.1.3 服务器上的文化设置	120
7.1.4 客户端的文化设置	121
7.1.5 翻译值和行为	122
7.2 资源文件的使用	126
7.2.1 本地资源	126
7.2.2 全局资源	130
第八章 用户个性化配置管理	133
8.1 使用个性化	134
8.1.1 配置个性化	134
8.1.2 定义个性化属性	136
8.1.3 使用个性化属性	138
8.1.4 使用个性化组	139
8.2 个性化和自定义数据类型	140
第九章 ASP.NET MVC 开发基础	142
9.1 MVC 简介	143
9.1.1 MVC 和 Web Form	143
9.1.2 ASP.NET MVC 的运行结构	143



9.2 ASP.NET MVC 基础	144
9.2.1 安装 ASP.NET MVC	144
9.2.2 新建一个 MVC 应用程序	145
9.2.3 ASP.NET MVC 应用程序的结构	146
9.2.4 运行 ASP.NET MVC 应用程序	149
9.3 ASP.NET MVC 原理	150
9.3.1 ASP.NET MVC 运行流程	150
9.3.2 ASP.NET MVC 工作原理	151
9.4 ASP.NET MVC 开发	154
9.4.1 创建 ASP.NET MVC 页面	154
9.4.2 ASP.NET MVC 数据呈现 (ViewData)	157
9.4.3 ASP.NET MVC 跨页数据呈现 (TempData)	159
9.4.4 ASP.NET MVC 页面重定向	162
9.4.5 ASP.NET MVC URL 路由 (URLRouting)	163
9.4.6 ASP.NET MVC 控件辅助工具 (Helper)	165
9.4.7 ASP.NET MVC 表单传值	167

上机部分

第一章 母版页 MasterPage	175
1.1 指导	176
1.2 练习	178
1.3 作业	178
第二章 使用导航控件	179
2.1 指导	180
2.1.1 创建站点地图 (WebSiteMap)	180
2.1.2 使用 SiteMapPath 控件	180
2.1.3 使用 Menu 控件	181
2.1.3.1 绑定到站点地图	181
2.1.3.2 绑定到 XML 文件	181
2.2 练习	183
2.3 作业	183
第三章 主题	186
3.1 指导	187
3.2 练习	190

3.3 作业	190
第四章 安全性与程序调试	192
4.1 指导	193
4.1.1 实现基于 Windows 验证用户组的方法	193
4.1.2 实现基于窗体的身份验证——根据 Web.config 中的值验证	193
4.2 练习	194
4.3 作业	194
第五章 成员和角色管理	195
5.1 上机指导(2 小时)	196
5.1.1 创建用户及用户登录	196
5.1.1.1 修改 Web.Config	196
5.2 作业	202
第六章 模块和处理程序	205
6.1 指导(2 小时)	206
6.1.1 使用 HttpModule	206
6.1.1.1 使用 URL 映射	206
6.1.1.2 使用 HttpModule 实现 URL 映射	207
6.1.2 使用 HttpHandler	209
6.1.2.1 使用 HttpHandle	209
6.2 作业	210
第七章 国际化与本地化	211
7.1 指导	212
7.1.1 实现国际化与本地化	212
7.2 练习	216
7.3 作业	216
第八章 个性化配置管理	218
8.1 上机指导	219
8.1.1 Profile 的存取示例——保留用户喜爱的网址	219
8.2 练习	220
8.3 作业	220
第九章 ASP.NET MVC 开发基础	222
9.1 指导	223
9.1.1 显示用户列表	223
9.2 练习	230
9.3 作业	230

第一章 理论部分

学习目标

◆ 理解，描述网站工作原理

◆ 掌握，利用网页制作创建界面网站

知识准备

◆ 完成一个具有三样页面的网站，你能够识别网页中的各种元素

◆ 如果某些网页的元素需要修改，你能够找到并修改它们

◆ 能思考一下，这些门户网站是如何设计的，它们有什么共同点

第一章 母版页 MasterPage

◆学习目标

- ◇ 理解：母版页的工作原理
- ◇ 掌握：利用母版页创建界面风格一致的网站

◆课前准备

- ◇ 完成一个具有三张页面的网站，你将如何保证它们的风格一致？
- ◇ 如果某些公用的元素需要修改，你能想到最简便的解决方案是什么？
- ◇ 请思考一下那些门户网站是如何保持每个页面都有一致风格的？

【本章简介】

目前,大多数 Web 站点在整个应用程序或应用程序的大多数页面中都有一些公共元素。比较下面两张东方网的专题页面,如图 1-1 所示。

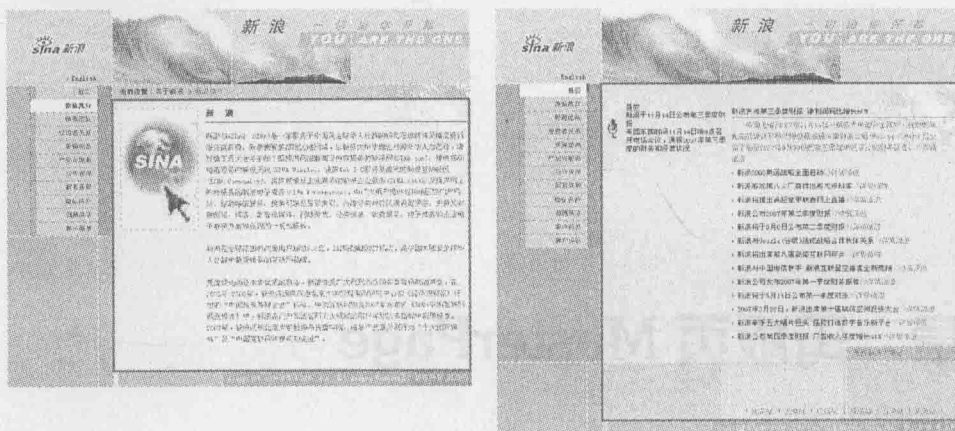


图 1-1

从大体上看,它们都包含一个有着丰富导航信息的开头,和一个显示版权声明的尾部。如果在每张页面分别实现这些内容,对这样一个门户网站来说重复的工作量是相当可观的。更可怕的是,随着网站运行时间的增长,内容的增加,一旦要对其中某些信息进行修改,这将是一场灾难性的工作。

后来尽管出现一些方案对解决此类问题有所改进,但 MasterPage 的出现才是最简单完美的解决方案。掌握它可以大大提高我们开发和维护的效率,下面让我们开始体验 MasterPage 的威力。

1.1 MasterPage 基础

MasterPage 用于提供一个模板,该模板可以由应用程序中任意多个 ASP.NET 页面使用。首先创建一个母板文件(*.master),除了显示众多页面公用的元素外,还包含一个或多个内容区域;以该母板文件为模板的页面文件(.aspx),主要用来在内容区域内显示页面信息,如图 1-2 所示。

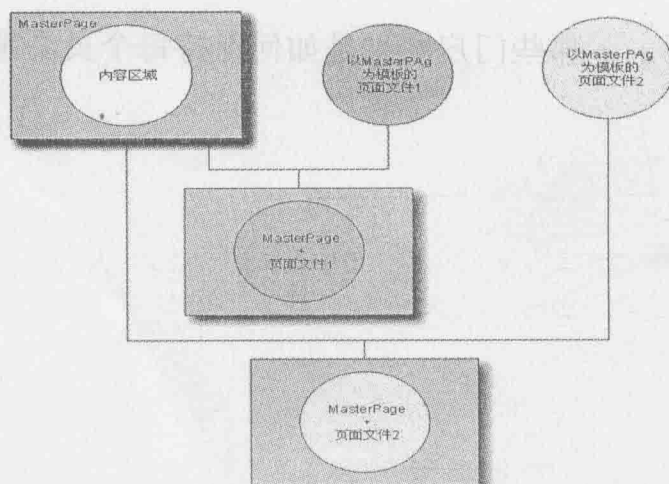


图 1-2

使用 Master 页面的一个优点是,在创建内容页面时,可以在 IDE 中看到模板。若在处理页面时可以看到

整个页面，就很容易开发出使用模板的内容页面。在处理内容页面时，所有模板项都呈灰色，不能编辑。可以修改的项会清晰地显示在模板中。这些可处理的区域称为内容区域，最初是在 Master 页面中定义的，如图 1-3 所示。

在 Master 页面中，指定了内容页面可以使用的区域。在 Master 页面中还可以有多个内容区域。图 1-4 显示了带两个内容区域的 Master 页面。

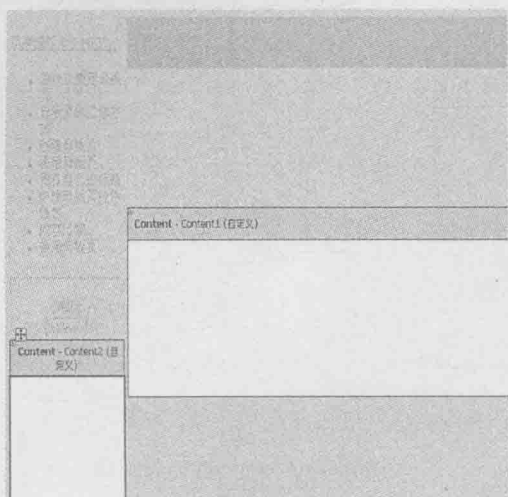


图 1-3



图 1-4

1.2 编写 MasterPage

1.2.1 给应用程序添加 MasterPage

在“添加新项”对话框中选择“母版页”，如图 1-5 所示。勾选“将代码放在单独的文件中”，点击“添加”按钮后，应用程序中会增加一个 MasterPage.master 页面和一个相关的 MasterPage.mater.cs 文件。

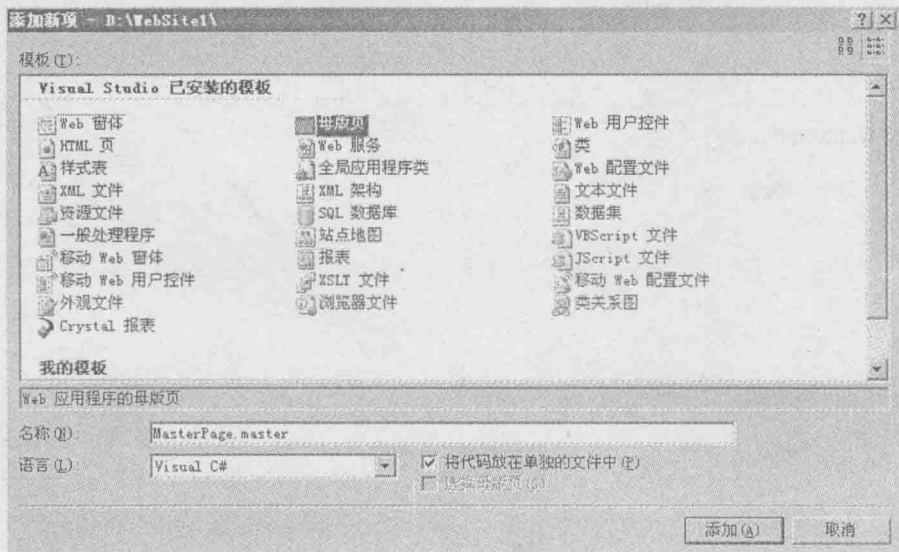


图 1-5



1.2.2 编写 MasterPage 上的公用代码

编写 MasterPage.Master 的内容如代码 1.1 所示:

代码 1.1

```
<%@ Master Language="C#" AutoEventWireup="true" CodeBehind="MasterPage.master.cs" Inherits="S3N_MasterPage.MasterPage" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" >

<head runat="server">

<title>迅腾教育 ASP.net2.0</title>

<style type="text/css">

    .style1

    { text-align:center }

    .style4

    { width: 181px; height: 80px; }

    .style6 {

        width: 181px;

        height: 46px;

    }

    .style8

    { width: 181px; height: 72px; }

    .style9

    {

        width: 179px;

        height: 46px;

    }

    .style10

    {

        width: 179px;

        height: 80px;

    }

    .style11

    {
```




```
</asp:ContentPlaceHolder>

</td>

</tr>

<tr>

<td class="style1" colspan="2" rowspan="2">

    &nbsp;

    &nbsp;

    这里是版权信息</td>

</tr>

</table>

</div>

</form>

</body>

</html>
```

首先看第一行代码：<%@ Master Language="C#" %>

这里不像一般的.aspx 页面那样使用 Page 指令，而是使用 Master 页面的 Master 指令。使用了一个属性 Language。本例中 Language 属性的值是 C#。

可以像编写其他.aspx 页面那样编写 Master 页面的其他部分，可以使用服务器控件、原始的 HTML 和文本、图像、事件或其他用于.aspx 页面的内容。也就是说，Master 页面也可以有 Page_Load 事件和其他事件。

注意

在代码 1.1 中，使用了一个新服务器控件<asp:ContentPlaceHolder>。这个控件定义了模板的区域，内容页面可以放置在该区域其内容在这个 Master 页面中，包含导航区域，广告区域和版权区域，页面中还定义了两个区域，任何继承该页面的内容页面都可以在这两个区域中放置其内容。

1.3 编写内容页面

应用程序中有了 Master 页面后，就可以把这个新模板用于应用程序的内容页面。在“解决方案资源管理器”中右击应用程序，选择“添加新项”，在应用程序中创建一个新的内容页面，输入页面名称，记得勾选上“选择母版页”，如图 1-6 所示。点击“添加”按钮后在随即弹出的“选择母版页”对话框中选一个母板作为该内容页面的模板，如图 1-7 所示。“确定”完成后便生成了一张与该母版页相关联的内容页面。

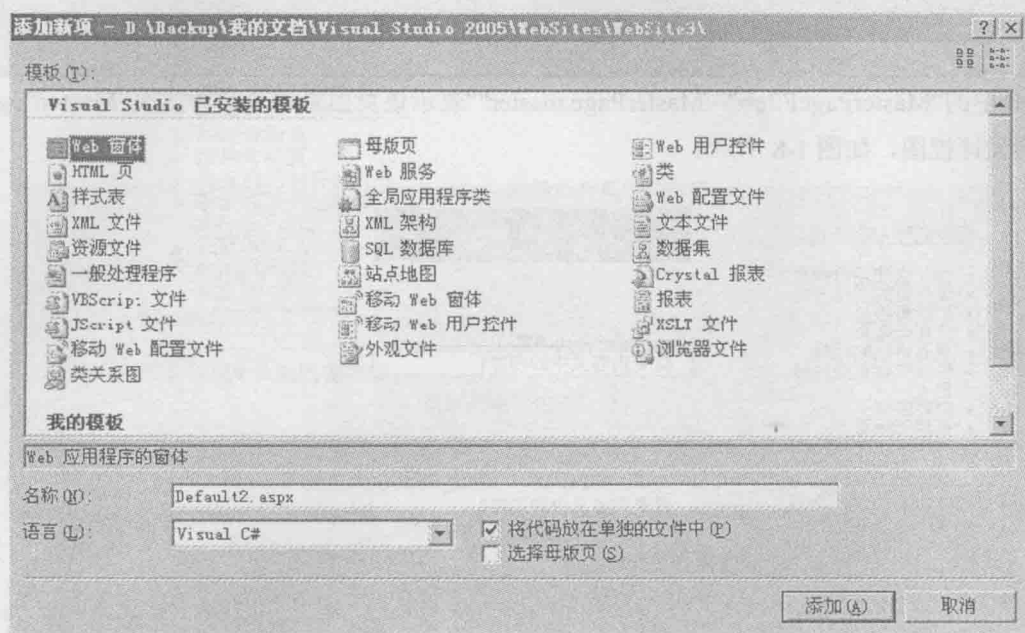


图 1-6

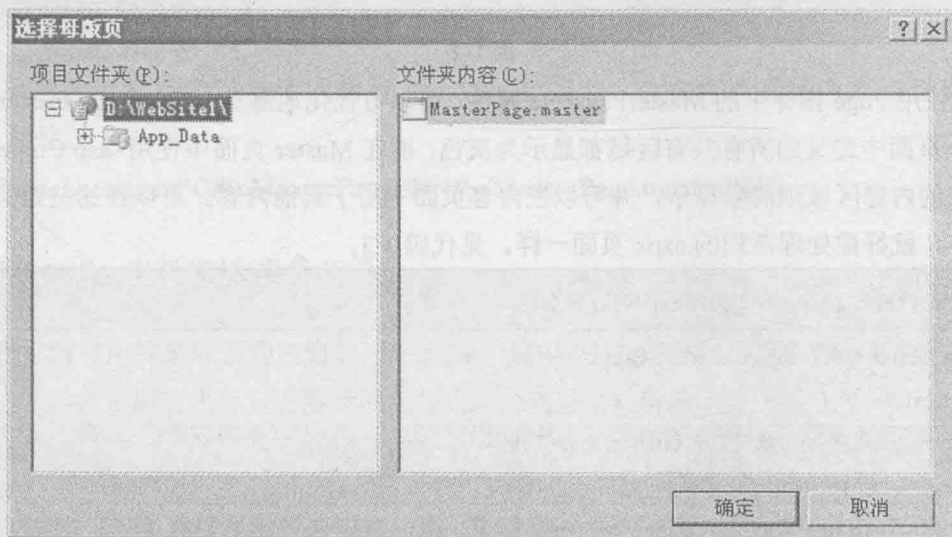


图 1-7

看一下内容页面的代码 1.2:

代码 1.2

```
<%@ Page Language="C#" MasterPageFile="~/MasterPage.Master" AutoEventWireup="true" CodeBehind="Default.aspx.cs" Inherits="S3N_MasterPage_Default" %>

<asp:Content ContentPlaceHolderID="ContentPlaceHolder1" runat="server">
    这是内容页中的内容 1
</asp:Content>

<asp:Content ContentPlaceHolderID="ContentPlaceHolder2" runat="server">
    这是内容页中的内容 2
```