

<<ASP.NET程序设计案例教程>>

图书基本信息

书名：<<ASP.NET程序设计案例教程>>

13位ISBN编号：9787115248275

10位ISBN编号：7115248273

出版时间：2011-4

出版时间：人民邮电出版社

作者：杨树林，胡洁萍 编著

页数：316

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<ASP.NET程序设计案例教程>>

内容概要

asp.net是微软公司推出的全新的互联网应用程序开发技术，是当今最主流的web程序开发技术之一。

《asp.net程序设计案例教程》共分9章，内容包括asp.net概述、c#语言基础、asp.net网页、用户界面设计、数据库访问技术、状态管理与数据缓存、asp.net常用技术、linq数据库技术和bbs综合案例。每章内容都与案例相结合，有助于学生理解知识，应用知识，达到学以致用。书中引进一些新知识和新方法，内容实用，重点突出，讲解精练，案例典型，既方便学习，又便于应用。

本书内容丰富，实例典型，知识讲解系统，可作为大中专院校计算机及其相关专业的教材，也可供软件开发人员及其他有关人员学习参考。

书籍目录

第1章 asp.net概述

1.1 asp.net简介

1.1.1 .net技术简介

1.1.2 什么是asp.net

1.1.3 .net 3.5

1.2 asp.net开发环境

1.2.1 安装visual studio 2008

1.2.2 创建web项目

1.2.3 管理web项目中的资源

1.2.4 创建asp.net网页

1.3 asp.net应用程序的构成

1.3.1 文件类型

1.3.2 文件夹类型

1.3.3 网站全局文件

1.3.4 asp.net配置

1.4 asp.net应用中的分层架构

1.4.1 分层架构模式

1.4.2 asp.net中的3层结构

1.4.3 案例1-1 网络书城系统

分析与设计

本章小结

习题与实验

第2章 c#语言基础

2.1 c#语言的基本语法

2.1.1 基本编码规则

2.1.2 数据类型

2.1.3 案例2-1 为书城网站定义用户权限枚举类型

2.1.4 运算符和表达式

2.2 字符串、日期和时间

2.2.1 字符串

2.2.2 datetime和timespan

2.2.3 案例2-2 日期操作工具类设计

2.2.4 数据类型的转换

2.3 流程控制与异常处理

2.3.1 分支结构

2.3.2 循环语句

2.3.3 异常处理

2.4 c#面向对象编程

2.4.1 类和对象

2.4.2 类的成员

2.4.3 继承

2.4.4 案例2-3 网络书城中的实体模型类设计

2.4.5 抽象类、接口与多态性

2.4.6 案例2-4 网络书城中的接口设计

2.5 数组和集合

<<ASP.NET程序设计案例教程>>

2.5.1 声明与访问数组

2.5.2 集合

2.5.3 案例2-5 网络书城中的购物车类设计

2.6 c# 3.5的新特征

2.6.1 隐型局部变量

2.6.2 扩展方法

2.6.3 lambda表达式

2.6.4 对象和集合初始化

2.6.5 匿名类型

本章小结

习题与实验

第3章 asp.net网页

3.1 概述

3.1.1 asp.net网页及其存储模式

3.1.2 asp.net网页生命周期

3.1.3 内置对象

3.2 asp.net web服务器控件

3.2.1 asp.net服务器控件类型

3.2.2 web服务器控件概述

3.2.3 常用的web控件

3.2.4 案例3-1 图书反馈网页的设计

3.2.5 验证控件

3.2.6 案例3-2 实现图书反馈网页的数据验证

3.3 页面切换与数据传递

3.3.1 页面切换

3.3.2 页面间的数据传递

本章小结

习题与实验

第4章 用户界面设计

4.1 主题

4.1.1 概述

4.1.2 创建主题

4.1.3 应用主题

4.1.4 案例4-1 书城网站的主题设计

4.2 母版页

4.2.1 母版页的基础知识

4.2.2 创建母版页和内容页

4.2.3 内容页和母版页的交互

4.2.4 案例4-2 书城网站的母版页设计

4.3 用户控件

4.3.1 用户控件简介

4.3.2 用户控件的创建

4.3.3 用户控件的使用

4.3.4 案例4-3 书城网站的用户控件设计

4.4 网站地图与页面导航

4.4.1 网站地图

4.4.2 使用导航地图实现网站导航

<<ASP.NET程序设计案例教程>>

4.4.3 案例4-4 书城的网站站点导航设计

本章小结

习题与实验

第5章 数据库访问技术

5.1 数据库访问基础

5.1.1 ado.net简介

5.1.2 数据库的连接

5.1.3 案例5-1 连接书城数据库

5.1.4 数据更新操作

5.1.5 数据查询操作

5.2 应用程序结构与数据操作

5.2.1 两层应用结构

5.2.2 三层应用结构

5.2.3 案例5-2 书城网站的dal层与bll层实现

5.3 数据绑定与数据绑定控件

5.3.1 数据绑定简介

5.3.2 gridview控件

5.3.3 detailsview控件与formview控件

5.3.4 案例5-3 实现书城网站的图书管理

5.4 其他数据绑定控件

5.4.1 datalist控件与repeater控件

5.4.2 案例5-4 实现书城网站图书分类菜单

5.4.3 listview控件与datapager控件

5.4.4 案例5-5 实现书城网站的主界面

本章小结

习题与实验

第6章 状态管理与数据缓存

6.1 asp.net状态管理概述

6.1.1 什么是状态管理

6.1.2 状态管理的类型

6.2 基于客户端的状态管理

6.2.1 视图状态

6.2.2 控件状态

6.2.3 隐藏域

6.2.4 cookie

6.2.5 查询字符串

6.2.6 案例6-1 完善书城网站用户登录程序

6.3 基于服务器的状态管理

6.3.1 应用程序状态

6.3.2 session状态管理

6.3.3 案例6-2 实现书城网站的用户统计

6.4 数据缓存

6.4.1 缓存概述

6.4.2 页输出缓存

6.4.3 使用应用程序缓存

6.4.4 缓存依赖

6.4.5 案例6-3 在书城网站中应用缓存技术

<<ASP.NET程序设计案例教程>>

本章小结

习题与实验

第7章 asp.net常用技术

7.1 成员资格与角色管理

7.1.1 验证方式及其配置

7.1.2 成员资格管理及其配置

7.1.3 asp.net登录控件

7.1.4 案例7-1 基于成员资格管理实现书城网站用户管理

7.2 个性化用户服务

7.2.1 个性化服务简介

7.2.2 个性化服务配置

7.2.3 个性化数据操作

7.2.4 为匿名用户实现个性化服务

7.2.5 案例7-2 实现网络书城购物车功能

7.3 验证码功能实现

7.3.1 绘图的基本知识

7.3.2 案例7-3 实现书城网站验证码

7.4 asp.net ajax

7.4.1 asp.net ajax概述

7.4.2 创建ajax应用

7.4.3 案例7-4 在书城网站中使用asp.net ajax

7.5 文件操作

7.5.1 文件的管理

7.5.2 文件的i/o操作

7.5.3 文件上传

7.5.4 案例7-5 实现书城网站的图书添加

本章小结

习题与实验

第8章 linq数据库技术

8.1 linq概述

8.1.1 什么是linq

8.1.2 基本的查询操作

8.1.3 linq查询表达式

8.1.4 使用linq进行数据转换

8.2 linq to ado.net

8.2.1 创建对象模型

8.2.2 查询和更改数据库

8.2.3 案例8-1 使用linq实现书城网站的数据访问层

8.2.4 存储过程

8.2.5 案例8-2 使用linq实现书城网站的结账

8.3 linq to xml

8.3.1 linq to xml概述

8.3.2 创建xml树

8.3.3 序列化xml树

8.3.4 查询xml树

8.3.5 修改xml树

本章小结

<<ASP.NET程序设计案例教程>>

习题与实验

第9章 bbs综合案例

9.1 系统分析与设计

9.1.1 系统分析

9.1.2 总体设计

9.1.3 创建对象模型

9.1.4 接口设计

9.2 数据访问层实现

9.2.1 版块数据访问类

9.2.2 主题数据访问类

9.2.3 帖子数据访问类

9.3 业务逻辑层实现

9.3.1 版块业务逻辑类

9.3.2 主题业务逻辑类

9.3.3 帖子业务逻辑类

9.4 系统配置

9.4.1 连接字符串及环境变量配置

9.4.2 验证模式、成员及角色管理配置

9.5 表现层设计

9.5.1 主题设计

9.5.2 母版及主页设计

9.5.3 主题视图设计

9.5.4 帖子视图设计

9.5.5 用户视图设计

本章小结

习题与实验

章节摘录

版权页：插图：页输出缓存在内存中存储处理后的ASENET页的内容。

这一机制允许ASP.NET向客户端发送页响应，而不必再次经过页处理生命周期。

页输出缓存对于那些不经常更改，但需要大量处理才能创建的页特别有用。

例如，如果创建大通信量的网页来显示不需要频繁更新的数据，页输出缓存则可以极大地提高该页的性能。

可以分别为每个页配置页缓存，也可以在Web.config文件中定义缓存配置，只定义一次缓存设置就可以在多个页中使用这些设置。

页输出缓存提供了两种页缓存模型：整页缓存和部分页缓存。

整页缓存允许将页的全部内容保存在内存中，并用于完成客户端请求。

部分页缓存允许缓存页的部分内容，其他部分则为动态内容。

部分页缓存可采用两种工作方式：控件缓存和缓存后替换。

控件缓存有时也称为分段缓存，这种方式允许将信息包含在一个用户控件内，然后将该用户控件标记为可缓存的，以此来缓存页输出的部分内容。

这一方式可缓存页中的特定内容，并不缓存整个页，因此每次都需重新创建整个页。

例如，如果要创建一个显示大量动态内容（如股票信息）的页，其中有些部分为静态内容（如每周总结），这时可以将静态部分放在用户控件中，并允许缓存这些内容。

缓存后替换与控件缓存正好相反。

它对页进行缓存，但是页中的某些片段是动态的，因此不会缓存这些片段。

例如，如果创建的页在设定的时间段内完全是静态的（如新闻报道页），可以设置为缓存整个页。

如果为缓存的页添加旋转广告横幅，则在页请求之间，广告横幅不断变化。

然而，使用缓存后替换，可以对页进行缓存，但可以将特定部分标记为不可缓存。

应用程序缓存应用程序缓存提供了一种编程方式，可通过键/值对将任意数据存储在内存中。

使用应用程序缓存与使用应用程序状态类似。

但是，与应用程序状态不同的是，应用程序缓存中的数据是易失的，即数据并不是在整个应用程序生命周期中都存储在内存中。

使用应用程序缓存的优点是由ASENET管理缓存，它会在项过期、无效或内存不足时移除缓存中的项。

还可以配置应用程序缓存，以便在移除项时通知应用程序。

使用应用程序缓存的模式是，确定在访问某一项时该项是否存在于缓存中，如果存在，则使用。

如果该项不存在，则可以重新创建该项，然后将其放回缓存中。

这一模式可确保缓存中始终有最新的数据。

<<ASP.NET程序设计案例教程>>

编辑推荐

《工业和信息化普通高等教育"十二五"规划教材立项项目:ASP.NET程序设计案例教程》涵盖主流技术和方法,体现教学需求;融合典型案例和知识,适应案例教学;突出项目引导和驱动,贴近实际应用。
内容实用,重点突出,讲解精练,案例典型。

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>