



XHTML

实践教程

(第2版)

- 构思独特，所有案例来自一线实战场景；
- 实用性强，将抽象的理论结合到实战案例上；
- 内容全面，结构清晰，体例丰富；
- 视频教学，专业教学视频帮助读者快速上手。

◎ 田中雨 郭磊 编著



清华大学出版社



XML 实践教程

(第2版)

◎ 田中雨 郭磊 编著

清华大学出版社
北京

内 容 简 介

在本书中详细介绍了 XML 技术知识。全书共分 14 章,包括 XML 技术优势、常用场合、发展前景;XML 文档的组成、声明、注释、处理指令、标记、标记属性、元素、非法的 PCDATA 字段、命名空间;与 XML 有关的 DTD 技术、XML Schema 文档、Web 服务、XPath 技术、XQuery 技术、XLink 技术、XPointer 技术、文档显示技术;以及处理 XML 文档数据时常用的 XML DOM 解析器和 SAX 解析器。本书中还介绍了 XML 与 RSS、站点地图、数据交换、XML 数据库以及 XML 技术在 C# 中的如何应用。最后,通过两个综合案例介绍与 XML 技术的使用。

本书既适用于学习 XML 技术的初学者使用,也适用于任何对 XML 用法有兴趣的程序员和专门从事与 XML 技术相关的人员使用。如果读者具有一定的 Web 编程或数据转换经验阅读起来会更加容易。

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 13701121933

图书在版编目(CIP)数据

XML 实践教程/田中雨,郭磊编著. —2 版. —北京:清华大学出版社,2016
(清华电脑学堂)

ISBN 978-7-302-40413-2

I. ①X… II. ①田… ②郭… III. ①可扩展语言-程序设计-教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2015)第 122575 号

责任编辑:夏兆彦
封面设计:张 阳
责任校对:徐俊伟
责任印制:王静怡

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社总机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者:北京富博印刷有限公司

装 订 者:北京市密云县京文制本装订厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:25.5 字 数:634 千字

版 次:2008 年 8 月第 1 版 2016 年 2 月第 2 版 印 次:2016 年 2 月第 1 次印刷

印 数:1~3500

定 价:49.00 元

产品编号:055137-01

XML 是使用 SGML 精简制作,并依照 HTML 的发展经验发展的一套使用规则严谨、数据描述简单的语言。它是一项用于 Web 开发的新技术,利用它可以通过 Internet 或者 Intranet 创建并发布信息。XML 具有自描述数据、数据重用、数据和表示分离,以及能够结构化和集成数据等优势,非常利于读者进行学习和创建,使用 XML 代码不仅清晰,而且容易理解。

本书是一本典型的 XML 实践教程,由多位经验丰富的编程人员编著而成。在本书中首先介绍 XML 的基础概念和语法规则,接着介绍与 XML 有关的其他技术,如 DTD、XSD、DOM、XLink 和 XPointer 等,最后通过两个章节的案例综合介绍 XML。

本书内容

本书以目前主流的 XML 技术为例进行介绍,可能还会涉及到 Java、C#和 JavaScript 等技术。全书共分为 14 章,主要内容如下:

第 1 章 XML 入门基础。本章主要介绍 XML 的基础知识,包括它的产生背景、技术优势、与 HTML 的区别、常用场合以及发展前景等内容。

第 2 章 XML 基础语法。本章详细介绍编写 XML 文件时的有关内容,依次介绍了 XML 文档的组成、声明、注释、处理指令、标记、元素、标记属性、非法的 PCDATA 字段以及命名空间等知识。

第 3 章 XML 与 DTD。本章首先介绍 DTD 的概念和构成,然后介绍 DTD 文件中的元素和属性,接着介绍了 XML 如何引用 DTD 文件,最后介绍实体的声明以及引用。

第 4 章 XML Schema 文档。本章首先介绍 XML Schema 文档的概念和特点,然后依次介绍文件的基本结构、常用的数据类型、注释、批注和其他空间的属性引用等内容。

第 5 章 XML 与 Web 服务。本章介绍 XML 与 Web 服务之间的联系,并且重点介绍 Web 服务的描述语言——WSDL。

第 6 章 XPath 和 XQuery 技术。本章首先简单介绍了 XSL 的知识,然后分别介绍了 XPath 技术和 XQuery 的内容,这两项技术实现了 XML 节点的定位和数据查询显示。

第 7 章 XLink 和 XPointer。本章首先介绍 XLink 技术的概念、语法、属性以及链接类型等,然后介绍了 XPointer 技术,包括该技术的概念、形式、对 XPath 技术的扩展等。每一项技术完成后都会通过一个实验指导实现一个小案例。

第 8 章 文档显示技术。本章详细介绍如何使用 XSL 的一个分支 XSLT 技术处理 XML 文档。包括 XSLT 的转换原理、创建和应用、与 CSS 的区别、XSLT 的常用模板、使用语法和常用函数等知识。

第 9 章 XML DOM 解析器。本章介绍如何使用 XML DOM 解析器处理 XML 文档中的数据。包括常用的解析器接口介绍、DOM 接口对象、DOM 的解析和加载、节点的

常用操作和节点类型等内容。

第 10 章 SAX 解析器。本章着重介绍如何使用 SAX 解析器处理 XML 文档中的数据。包括 SAX 的发展历史、SAX 的创建和 XML 解析、核心接口和扩展接口、基本应用以及优点和缺点等内容。

第 11 章 XML 的典型应用。XML 以其兼容性和可扩展性被广泛应用，本章将介绍 XML 在网络和程序中的应用，包括 XML 与 RSS、XML 与 SiteMap 以及 XML 与 SQL 等内容。

第 12 章 XML 在 C# 中的应用。XML 可以单独在开发工具中使用，也可以在 Java 语言开发的平台上使用，还可以在 C# 语言的开发平台上使用。本章介绍如何在 Visual Studio 2010 中使用 C# 语言处理与 XML 文档有关的内容。

第 13 章 XML 新闻管理系统。本章的综合案例以 Visual Studio 2010 工具为开发平台，以 XML 文档作为数据库实现新闻管理系统前台新闻的添加、遍历、修改、删除、查看列表和详细信息的功能。

第 14 章 商品信息管理系统。本章以 XML 文件存储网购信息和团购信息，综合本书的知识点，通过 ASP.NET、C# 语言与 XML 的结合，介绍商品信息管理系统的使用。

本书特色

本书中的大量内容来自实际的开发项目，而且本书针对初、中级读者由浅入深地介绍与 XML 有关的知识。本书具有以下特色：

□ 知识全面，内容丰富

本书围绕 XML 文档和有关的技术展开讲解，涵盖了实际开发中使用 XML 处理比较复杂内容的代码。

□ 基于理论，注重实践

本书不仅介绍理论知识，而且在合适位置安排综合应用实例，或者小型应用程序，将理论知识应用到实践中，加强读者实际应用能力，巩固系统基础知识。另外，还将一些概念和术语放到文档中，以方便读者了解。

□ 应用广泛，提供文档

对于大多数的精选案例，都向读者提供了详细步骤、结构清晰简明，分析深入浅出，而且有些程序能够直接在项目中使用，避免读者进行二次开发。

□ 视频教学

本书为实例配备了视频教学文件，读者可以通过视频文件更加直观地学习与 XML 有关的技术知识。所有视频教学文件均已上传到 www.ztydata.com.cn，读者可自行下载。

□ 网站技术支持

读者在学习或者工作的过程中，如果遇到实际问题，可以直接登录 www.itzcn.com 与我们取得联系，作者会在第一时间给予帮助。

读者对象

本书适合作为学习 XML 技术的入门程序开发人员的自学用书，也适合作为高等院

校相关专业的教学参考书，还可以供开发人员查阅和参考。

- 学习 XML 技术的入门者。
- XML 技术的初级学者以及在校学生。
- 各大中专院校的在校学生和相关授课老师。
- 准备从事与 XML 技术相关的人员。

除了封面署名人员之外，参与本书编写的人员还有李海庆、王咏梅、康显丽、王黎、汤莉、倪宝童、赵俊昌、康显丽、方宁、郭晓俊、杨宁宁、王健、连彩霞、丁国庆、牛红惠、石磊、王慧、李卫平、张丽莉、王丹花、王超英、王新伟等。本书在编写过程中难免会有漏洞，欢迎读者通过清华大学出版社网站 www.tup.tsinghua.edu.cn 与我们联系，帮助我们改正提高。

编 者

| | | | |
|--------------------------------------|----|-------------------------------------|----|
| 第 1 章 XML 入门基础 | 1 | 2.1.4 XML 文档的处理指令 | 23 |
| 1.1 XML 的概念 | 1 | 2.2 XML 标记概述 | 24 |
| 1.1.1 认识 XML | 1 | 2.2.1 标记的意义 | 24 |
| 1.1.2 发展历史 | 3 | 2.2.2 标记的命名 | 25 |
| 1.1.3 技术优势 | 3 | 2.3 XML 元素概述 | 25 |
| 1.2 XML 和 HTML 的区别 | 4 | 2.3.1 元素的概念 | 25 |
| 1.2.1 认识 SGML | 5 | 2.3.2 元素的分类 | 26 |
| 1.2.2 与 HTML 的区别 | 5 | 2.3.3 元素的命名 | 27 |
| 1.3 XML 开发工具 | 6 | 2.3.4 元素特性 | 28 |
| 1.4 XML 标准的诞生 | 7 | 2.4 XML 标记属性概述 | 28 |
| 1.4.1 了解 W3C | 7 | 2.4.1 属性的使用 | 28 |
| 1.4.2 XML 标准 | 8 | 2.4.2 注意事项 | 29 |
| 1.4.3 与 XML 相关的标准 | 8 | 2.4.3 元素和属性的转换 | 30 |
| 1.5 XML 的常用场合 | 10 | 2.4.4 属性的使用 | 31 |
| 1.5.1 常用的应用领域 | 10 | 2.5 非法的 PCDATA 字段 | 32 |
| 1.5.2 数据交互 | 11 | 2.5.1 字符和实体引用 | 33 |
| 1.5.3 Web 服务 | 11 | 2.5.2 CDATA 文本段 | 35 |
| 1.5.4 Web 集成 | 12 | 2.6 XML 命名空间 | 36 |
| 1.5.5 内容管理 | 12 | 2.6.1 命名空间概述 | 36 |
| 1.5.6 配置 | 12 | 2.6.2 自定义命名空间 | 36 |
| 1.6 XML 的发展前景 | 13 | 2.6.3 默认的命名空间 | 38 |
| 1.6.1 商务的自动化处理 | 13 | 2.6.4 命名空间的注意事项 | 39 |
| 1.6.2 信息发布 | 13 | 2.7 XML 文档小结 | 40 |
| 1.6.3 智能化的 Web 应用程序和 数据集成 | 13 | 2.7.1 XML 文档的遵循规则 | 40 |
| 1.7 实验指导 1-1: 显示一个基本 XML 文档 | 13 | 2.7.2 XML 文档的两种形式 | 41 |
| 1.8 思考与练习 | 16 | 2.8 实验指导 2-1: 编写通讯录 XML 文档 | 41 |
| 第 2 章 XML 基础语法 | 18 | 2.9 思考与练习 | 43 |
| 2.1 XML 文档概述 | 18 | 第 3 章 XML 与 DTD | 45 |
| 2.1.1 XML 文档的组成 | 18 | 3.1 DTD 简介 | 45 |
| 2.1.2 XML 文档的声明 | 20 | 3.1.1 DTD 概述 | 45 |
| 2.1.3 XML 文档的注释 | 22 | 3.1.2 DTD 的构成 | 46 |
| | | 3.2 元素声明 | 48 |

| | | | | | |
|----------------------------|--------------------------|----|--------------------------------|--------------------------------|-----|
| 3.2.1 | 元素声明简介 | 48 | 4.4.3 | 引用其他空间的属性 | 95 |
| 3.2.2 | 元素声明分类 | 49 | 4.5 | 将 XML Schema 文档引入到 XML 中 | 95 |
| 3.2.3 | 元素指示符 | 51 | 4.6 | 来自多个文档的模式 | 96 |
| 3.3 | 属性 | 51 | 4.7 | 实验指导 4-1: 两种方法编写 XML Schema 文档 | 97 |
| 3.3.1 | 属性声明 | 51 | 4.8 | 思考与练习 | 103 |
| 3.3.2 | 属性类型 | 53 | 第 5 章 XML 与 Web 服务 | | 104 |
| 3.3.3 | 属性附加声明 | 57 | 5.1 | XML 与 Web 服务 | 104 |
| 3.4 | 实验指导 3-1: 研究生班级信息 | 59 | 5.1.1 | Web 服务 | 104 |
| 3.5 | DTD 引用 | 60 | 5.1.2 | WSDL 语言基础 | 107 |
| 3.5.1 | 引用内部 DTD | 60 | 5.1.3 | WSDL 文档编写 | 108 |
| 3.5.2 | 引用外部 DTD | 61 | 5.2 | 文档结构 | 111 |
| 3.5.3 | 混合引用 DTD | 62 | 5.2.1 | definitions 根元素 | 111 |
| 3.6 | 实体声明与引用 | 63 | 5.2.2 | types 元素 | 112 |
| 3.6.1 | 实体的概念与分类 | 63 | 5.2.3 | message 元素 | 114 |
| 3.6.2 | 内部普通实体 | 65 | 5.2.4 | portType 元素 | 115 |
| 3.6.3 | 外部普通实体 | 67 | 5.2.5 | binding 元素 | 117 |
| 3.6.4 | 参数实体 | 70 | 5.2.6 | service 元素 | 119 |
| 3.7 | 实验指导 3-2: 商品信息 DTD 与 XML | 72 | 5.3 | WSDL 技术 | 119 |
| 3.8 | 思考与练习 | 74 | 5.3.1 | WSDL 端口 | 119 |
| 第 4 章 XML Schema 文档 | | 75 | 5.3.2 | 使用自定义 WSDL | 121 |
| 4.1 | 了解 XML Schema | 75 | 5.3.3 | WSDL 文档使用原理 | 121 |
| 4.1.1 | XML Schema 的概念 | 75 | 5.3.4 | WSDL 调用 Web 服务 | 122 |
| 4.1.2 | XML Schema 的特点 | 76 | 5.4 | 实验指导 5-1: 两数相加 Web 服务 | 126 |
| 4.1.3 | XML Schema 的两种模式 | 77 | 5.5 | 思考与练习 | 129 |
| 4.1.4 | XML Schema 与 XML DTD | 78 | 第 6 章 XPath 和 XQuery 技术 | | 131 |
| 4.2 | XML Schema 的基本结构 | 78 | 6.1 | XSL 简介 | 131 |
| 4.2.1 | XML Schema 文档文件 | 78 | 6.2 | XPath 简介 | 134 |
| 4.2.2 | 创建根元素: schema | 79 | 6.3 | XPath 定位 | 136 |
| 4.2.3 | 创建元素: element | 81 | 6.3.1 | 路径表达式 | 136 |
| 4.2.4 | 声明 simpleType 元素 | 82 | 6.3.2 | XPath 轴 | 139 |
| 4.2.5 | 声明 complexType 元素 | 85 | 6.3.3 | 节点测试 | 143 |
| 4.2.6 | 声明属性元素: attribute | 91 | 6.3.4 | XPath 运算符 | 146 |
| 4.3 | 常用的数据类型 | 92 | 6.4 | XPath 函数 | 146 |
| 4.3.1 | 内置的数据类型 | 92 | 6.4.1 | 字符串函数 | 147 |
| 4.3.2 | 用户定义的数据类型 | 93 | 6.4.2 | 数值函数 | 148 |
| 4.4 | XML Schema 的其他内容 | 94 | 6.4.3 | 布尔函数 | 149 |
| 4.4.1 | 注释 | 94 | | | |
| 4.4.2 | 批注 | 94 | | | |

| | | | | | |
|--------------|--------------------------------|------------|--------------|--------------------|------------|
| 6.4.4 | 节点集函数 | 149 | 8.2.1 | 模板 | 201 |
| 6.5 | XQuery 简介 | 151 | 8.2.2 | 模板与标记匹配 | 202 |
| 6.6 | XQuery 表达式 | 151 | 8.2.3 | 调用模板 | 204 |
| 6.6.1 | XQuery 表达式基础 | 152 | 8.3 | XSLT 语法 | 206 |
| 6.6.2 | FLWOR 表达式 | 153 | 8.3.1 | XSLT 标记 | 206 |
| 6.7 | XQuery 函数 | 155 | 8.3.2 | 简单判断 | 207 |
| 6.8 | 实验指导 6-1: 团购信息管理 | 156 | 8.3.3 | 多条件判断 | 212 |
| 6.9 | 思考与练习 | 157 | 8.3.4 | 循环处理 | 214 |
| 第 7 章 | XLink 和 XPointer | 159 | 8.3.5 | 输出内容的排序 | 216 |
| 7.1 | 了解链接 | 159 | 8.4 | XLST 函数 | 218 |
| 7.1.1 | HTML 中的链接 | 159 | 8.4.1 | 函数 | 218 |
| 7.1.2 | XML 中的链接 | 160 | 8.4.2 | Current()函数 | 219 |
| 7.2 | 认识 XLink 技术 | 161 | 8.4.3 | document()函数 | 221 |
| 7.2.1 | XLink 介绍 | 161 | 8.4.4 | format-number()函数 | 221 |
| 7.2.2 | XLink 语法 | 161 | 8.4.5 | NaN 值 | 222 |
| 7.2.3 | XLink 属性 | 162 | 8.5 | 在 XSLT 中使用 CSS | 223 |
| 7.2.4 | 可选和必需属性 | 163 | 8.6 | 实验指导 8-1: 笔记本信息显示 | 225 |
| 7.3 | XLink 的链接类型 | 164 | 8.7 | 思考与练习 | 229 |
| 7.3.1 | 简单链接 | 164 | 第 9 章 | XML DOM 解析器 | 230 |
| 7.3.2 | 扩展链接 | 168 | 9.1 | 常用的解析器接口 | 230 |
| 7.3.3 | 完整的扩展链接 | 174 | 9.1.1 | 了解解析器 | 230 |
| 7.4 | 实验指导 7-1: XLink 将图像嵌入 链接 | 176 | 9.1.2 | DOM | 231 |
| 7.5 | 认识 XPointer 技术 | 178 | 9.1.3 | SAX | 231 |
| 7.5.1 | XPointer 介绍 | 179 | 9.1.4 | JDOM | 232 |
| 7.5.2 | XPointer 形式 | 180 | 9.1.5 | DOM4J | 232 |
| 7.5.3 | 位置路径与集合 | 182 | 9.2 | 了解 XML DOM | 233 |
| 7.5.4 | XPointer 对 XPath 的扩展 | 185 | 9.2.1 | W3C 与 DOM | 233 |
| 7.5.5 | XPointer 示例 | 189 | 9.2.2 | DOM 的文档树模型 | 234 |
| 7.6 | 实验指导 7-2: 使用 XSLT 转换 XML 文档 | 191 | 9.3 | DOM 接口对象 | 236 |
| 7.7 | 思考与练习 | 193 | 9.3.1 | 常用的 DOM 接口 | 236 |
| 第 8 章 | 文档显示技术 | 195 | 9.3.1 | 常用的 DOM 对象 | 238 |
| 8.1 | XSLT 简介 | 195 | 9.3.3 | 文档对象: Document | 238 |
| 8.1.1 | XSL 与 XSLT | 195 | 9.3.4 | 节点对象: Node | 239 |
| 8.1.2 | XSLT 转换显示原理 | 196 | 9.3.5 | 节点列表对象: NodeList | 240 |
| 8.1.3 | XSLT 文档创建与应用 | 199 | 9.4 | DOM 的解析和加载 | 240 |
| 8.1.4 | XSLT 与 CSS 比较 | 200 | 9.5 | 节点操作 | 243 |
| 8.2 | XSLT 模板 | 201 | 9.5.1 | 遍历节点 | 243 |
| | | | 9.5.2 | 添加节点 | 245 |
| | | | 9.5.3 | 删除节点 | 247 |

| | | | | | |
|-----------------------|--------------------------|-----|-----------------------------|------------------------------|-----|
| 9.5.4 | 改变节点 | 249 | 10.7 | 实验指导 10-1: 解析 XML 文档并输出信息 | 292 |
| 9.5.5 | 替换节点 | 250 | 10.8 | 思考与练习 | 295 |
| 9.5.6 | 克隆节点 | 252 | 第 11 章 XML 的典型应用 | 297 | |
| 9.6 | 节点类型 | 253 | 11.1 | XML 与 RSS | 297 |
| 9.6.1 | nodeType 属性取值 | 253 | 11.1.1 | RSS 简介 | 297 |
| 9.6.2 | 节点类型 | 254 | 11.1.2 | 使用 RSS | 300 |
| 9.7 | 实验指导 9-1: DOM 对象操作实现级联菜单 | 255 | 11.1.3 | RSS 元素 | 301 |
| 9.8 | 思考与练习 | 259 | 11.1.4 | RDF 简介 | 304 |
| 第 10 章 SAX 解析器 | | 261 | 11.1.5 | RDF 元素 | 305 |
| 10.1 | 了解 SAX 解析器 | 261 | 11.2 | 站点地图 | 306 |
| 10.1.1 | SAX 的工作机制 | 261 | 11.3 | 配置文件 | 307 |
| 10.1.2 | SAX 的发展历史 | 263 | 11.3.1 | Web.config 文件 | 308 |
| 10.1.3 | 与 DOM 解析器的区别 | 263 | 11.3.2 | web.xml 文件 | 312 |
| 10.2 | SAX 的创建和 XML 解析 | 264 | 11.4 | 数据交换 | 313 |
| 10.2.1 | JAXP 为 SAX 提供的 API | 264 | 11.4.1 | 数据交换的类型 | 314 |
| 10.2.2 | SAX 的创建 | 265 | 11.4.2 | 数据存取机制 | 315 |
| 10.2.3 | XML 文档解析 | 266 | 11.5 | XML 与数据库 | 316 |
| 10.3 | SAX 的有关接口 | 268 | 11.5.1 | XML 与关系数据库 | 316 |
| 10.3.1 | ContentHandler 接口 | 268 | 11.5.2 | XML 与面向对象的数据库 | 318 |
| 10.3.2 | DTDHandler 接口 | 269 | 11.5.3 | SQL Server 对 XML 的支持 | 319 |
| 10.3.3 | EntityResolver 接口 | 270 | 11.6 | XML 与 SQL Server 的数据交换 | 319 |
| 10.3.4 | ErrorHandler 接口 | 270 | 11.6.1 | 配置 IIS 服务器 | 320 |
| 10.3.5 | 扩展接口 | 270 | 11.6.2 | 通过 HTTP 方式访问 SQL Server | 321 |
| 10.4 | SAX 的应用 | 271 | 11.7 | 访问数据库 | 323 |
| 10.4.1 | 处理文件开始和结束 | 271 | 11.7.1 | 使用 HTTP 访问数据库对象 | 323 |
| 10.4.2 | 处理特殊指令 | 273 | 11.7.2 | 使用 ADO 实现数据交换 | 323 |
| 10.4.3 | 处理开始和结束标记 | 274 | 11.8 | 实验指导 11-1: 使用 ASP.NET 修改 XML | 324 |
| 10.4.4 | 处理文本数据 | 277 | 11.9 | 思考与练习 | 326 |
| 10.4.5 | 处理空白符 | 279 | 第 12 章 XML 在 C# 中的应用 | 327 | |
| 10.4.6 | 处理命名空间 | 280 | 12.1 | XmlDataSource 控件 | 327 |
| 10.4.7 | 处理实体 | 283 | 12.1.1 | XmlDataSource 控件概述 | 327 |
| 10.5 | SAX 的其他应用 | 287 | | | |
| 10.5.1 | 处理错误 | 287 | | | |
| 10.5.2 | 处理异常 | 289 | | | |
| 10.6 | SAX 的优点和缺点 | 290 | | | |
| 10.6.1 | SAX 的优点 | 290 | | | |
| 10.6.2 | SAX 的缺点 | 291 | | | |

| | | | | | |
|---------------|--|-----|---------------|-----------------------|-----|
| 12.1.2 | 使用 XmlDataSource 控件 | 328 | 13.3.1 | 首页效果 | 359 |
| 12.2 | XML 与 DataSet 的交互 | 330 | 13.3.2 | 实现代码 | 360 |
| 12.2.1 | 了解 DataSet 对象 | 330 | 13.4 | 查看具体新闻 | 363 |
| 12.2.2 | ReadXml()读取文件 | 331 | 13.5 | 添加新闻内容 | 365 |
| 12.2.3 | WriteXml()写入文件 | 335 | 13.5.1 | 添加页面 | 365 |
| 12.3 | System.Xml 命名空间 | 337 | 13.5.2 | 实现代码 | 365 |
| 12.4 | XmlReader 读取 XML 文件 | 340 | 13.6 | 编辑新闻内容 | 369 |
| 12.4.1 | XmlReader 的创建 | 340 | 13.6.1 | 编辑页面 | 369 |
| 12.4.2 | XmlReader 的属性 | 341 | 13.6.2 | 实现代码 | 370 |
| 12.4.3 | XmlReader 的方法 | 342 | 13.7 | 删除单条新闻 | 372 |
| 12.4.4 | XmlReader 的使用步骤 | 346 | 13.7.1 | 直接删除新闻 | 372 |
| 12.4.5 | XmlReader 与 SAX 比较 | 346 | 13.7.2 | 通过链接删除 | 374 |
| 12.5 | XmlWriter 写入 XML 文件 | 347 | 第 14 章 | 商品信息管理系统 | 376 |
| 12.5.1 | XmlWriter 的创建 | 347 | 14.1 | 系统概述 | 376 |
| 12.5.2 | XmlWriter 的方法 | 349 | 14.1.1 | 需求分析 | 376 |
| 12.5.3 | XmlWriter 的使用步骤 | 351 | 14.1.2 | 功能描述 | 377 |
| 12.6 | 实验指导 12-1: 数据库中的评论 保存为 XML 数据 | 351 | 14.2 | 数据设计 | 377 |
| 12.7 | 思考与练习 | 354 | 14.2.1 | DTD 设计 | 377 |
| 第 13 章 | XML 新闻管理系统 | 355 | 14.2.2 | XML 设计 | 378 |
| 13.1 | 新闻管理系统概述 | 355 | 14.3 | 页面设计 | 379 |
| 13.1.1 | 产生背景 | 355 | 14.4 | 典型代码实现 | 380 |
| 13.1.2 | 功能描述 | 356 | 14.4.1 | 团购信息查询显示 | 380 |
| 13.2 | 设计 XML 文件 | 358 | 14.4.2 | 团购信息添加 | 384 |
| 13.3 | 设计首页列表 | 359 | 14.4.3 | 团购信息删除 | 385 |
| | | | 14.4.4 | 团购信息修改 | 388 |
| | | | 14.4.5 | 其他典型代码实现 | 389 |
| | | | 附录 | 思考与练习答案 | 392 |

第 1 章 XML 入门基础

XML 在一直不断地快速发展，它正在成为一切 Web 服务和大多数 SOA（面向服务架构，Service-Oriented Architecture）的基础。XML 与 HTML 一样，都是一种标记语言，可以支持开发者为 Web 信息标记设计自己的标记，这样能够使 Web 信息被 XML 应用所共享。随着 XML 的发展，出现了许多与之相关的新技术，例如 XSLT、DTD、DOM 和 XPath 等。

本章中将详细介绍 XML 的入门基础知识，包括它产生的背景、有哪些优势、与 HTML 技术的区域、主要开发工具以及应用和前景等内容。通过对本章的学习，读者可以对 XML 有新的认识。

本章学习目标：

- 熟悉 XML 的概念和特点
- 掌握 XML 与 HTML 的区别
- 了解 XML 的开发工具有哪些
- 了解使用 XML 标准的原因
- 掌握与 XML 相关的标准
- 熟悉 XML 的应用领域
- 熟悉 XML 的发展前景

1.1 XML 的概念

标记语言通常会被称为置标语言，它是一种可以展现出关于文档结构和数据处理细节的文字编码。标记语言广泛应用于网页和网络应用程序，当今广泛使用的置标语言是超文本标记语言（HyperText Markup Language，HTML）和可扩展标记语言（eXtensible Markup Language，XML）。

在本节中的三个小节中将简单介绍 XML 的基本知识，包括概念、特点、发展历史和技术优势等内容。

1.1.1 认识 XML

XML 是英文 eXtensible Markup Language 的缩写，中文通常会被称为可扩展标记语言。XML 用于标记电子文件使其具有结构性的标记语言，可以用来标记数据、定义数据类型，是一种允许用户对自己的标记语言进行定义的源语言。它提供统一的方法来描述和交换独立于应用程序或供应商的结构化数据。

1. XML 的特点

XML 是 W3C 推荐参考通用标记语言，同样也是 SGML 的子类，可以定义自己的一组标记，主要特点如下所示。

- XML 是一种元语言，它描述的是结构和语义，而不是格式化。
- 允许通过使用自定义格式，标识、交换和处理数据库可以理解的数据。
- 基于文本的格式，允许开发人员描述结构化数据并在各种应用之间发送和交换这些数据。
- 有助于在服务器之间传输结构化数据。

2. XML 是元标记语言

XML 是一种元标记语言，用户可以定义自己需要的标记，这些标记必须根据某些通用的原理来创建。但是，在标记的意义上也具有相当的灵活性。假设用户正在处理与族谱有关的事情，那么需要描述某个人的出生、死亡、埋葬地点、家庭、结婚情况和离婚情况等，这就必须创建用于每一项的标记。针对上述描述项标记的创建，可以在文档类型定义（Document Type Definition, DTD）中加以描述。

3. XML 定义了元句法

XML 定义了一套元句法，与特定领域有关的标记语言（例如 MusicML、MathML 和 CML）都必须遵守。如果一个应用程序可以理解元句法，那么该程序也能自动地理解所有由此元语言建立起来的其他语言。浏览器不需要了解多种不同的标记语言使用每个标记，实际上，浏览器在读入文档或者是其他的 DTD 时才会了解给定文档使用的标记。

4. XML 可以用来做什么

使用 XML 可以在 Internet 上交换金融信息，在未来会看到 XML 和 B2B 的使用，XML 即将成为在 Internet 上企业之间交换金融信息的主要语言。那么 XML 到底能够做什么呢？

（1）XML 可以用来共享数据

使用 XML，纯文本文件可以用于共享数据，由于 XML 数据是用纯文本格式存储的，因此 XML 提供了一种独立于软件和硬件的共享数据方式。这样创建不同的应用程序都能够使用的数据就会更加容易，并且将系统扩展升级到一个新的操作系统、服务器、应用程序和新的浏览器也会更加容易。

（2）XML 可以用来存储数据

有了 XML，就可以用纯文本文件来存储数据。XML 可以用来在文件或数据库中存储数据，可以用普通的应用程序来显示数据。

（3）XML 可以使数据更有用

有了 XML，更多的用户可以使用数据。由于 XML 对于软件、硬件和应用程序都是相互独立的，因此可以使标准 HTML 浏览器以外的其他应用程序也能够使用数据。其他客户和浏览器可以将 XML 文件作为数据源来进行访问，就像访问数据库一样。数据对

于所有的“阅读机器”都可用，它可以使数据更容易被盲人或残疾人使用。

(4) XML 可以用来创建新的语言

XML 是 WAP 和 WML 之母，无线标记语言 (WML) 是用于标记手提设备 Internet 应用程序的，也是使用 XML 进行编写的。

1.1.2 发展历史

XML 的前身是 SGML (The Standard Generalized Markup Language)，它是自 IBM 从 20 世纪 60 年代开始发展的 GML (Generalized Markup Language) 标准化后的名称。

1978 年，ANSI 将 GML 加以整理规范，发布成为 SGML，1986 年起为 ISO 所采用 (ISO 8879)，并且被广泛地应用在各种大型的文件计划中，但是 SGML 是一种非常严谨的文件描述法，导致过于庞大复杂 (标准手册就有 500 多页)，难以理解和学习，进而影响其推广与应用。于是，人们对 SGML 进行了简化衍生出 HTML。HTML 在初期没有任何定义文档外观的相关方法，仅用于在浏览器里显示网页文件。后来，随着因特网的发展，人们为了控制文件样式，扩充了描述如何显示数据的标记。在 Netscape 与 Microsoft 之间的浏览器大战后，HTML 标准权威性遭受重大的考验。到 HTML 4.0 时，W3C 又恢复了其地位。同时 W3C 意识到 HTML 的缺陷：

- 不能解决所有解释数据的问题 例如，如果是影音文件或化学公式、音乐符号等其他型态的内容。
- 效能问题 需要下载整份文件，才能开始对文件做搜寻的动作。
- 扩充性、弹性、易读性均不佳。

1996 年 11 月，在美国波士顿的 SGML 年会上，为了解决以上问题，专家们使用 SGML 精简制作，并依照 HTML 的发展经验，发展出一套使用规则严谨，但数据描述简单的语言——XML。XML 就是在一个这样的背景下诞生的，它的目的是提供一个对信息能够做精准描述的机制，弥补 HTML 太过于表现导向的特点。

1996 年开始有了 XML 的雏形，并且向 W3C 进行提案，在 1998 年 2 月发布为 W3C 的标准 (XML1.0)。XML 精简了一大片 SGML 难得用一次的功能，正如几十万汉字中常用的只不过八千，SGML 常用的部分只占 20%，它抛弃了 SGML 中不常用的部分精简了 80%。

1.1.3 技术优势

XML 方便学习和创建，代码不仅清晰，而且便于阅读理解。XML 使许多只利用 HTML 不可能完成的任务得以完成。由于 XML 是可扩展的，开发者喜欢 XML 有许多原因，其主要技术优势如下所示。

1. 设计与特定领域有关的标记语言

用户可以使用 XML 自由地制定自己的标记语言，而不需要微软、Netscape、W3C。它允许各种不同的专业人士 (例如音乐家、化学家和数学学者等) 开发与自己的特定领

域有关的标记语言，使领域中的人们可以交换笔记、数据和信息，而不用担心接收端的人是否有特定的软件来创建数据。

特定领域的开发人员甚至可以向本领域外的人员发送文档，至少接受文档的人能够查看文档的内容。进一步来讲，为特别领域创建标记语言不会产生“病件”或是对于本专业外的人来说产生不必要的复杂性。

2. 自描述数据

XML 在基本水平上使用的是非常简单的数据格式，可以使用 100% 的纯 ASCII 文本来书写，也可以使用几种其他定义好的格式来书写。ASCII 文本是几乎不会“磨损”的，丢失一些字节甚至是相当多的字节，剩下的数据还是可以读取的。与许多格式形成了鲜明的对比，例如压缩数据或是串行的 Java 对象，这些数据即使丢失一个字节，剩余的数据也不能读取。

3. 数据重用

XML 是被设计用来存储数据、携带数据和交换数据的，它不是为了显示数据而设计的。一个存储数据的 XML 文档，可以被程序解析，把里面的数据提取出来加以利用，也可以被放到数据库中，也可以通过网络传输到另外一台计算机上，被解析使用。这些数据可以在多种场合被使用和调用。

4. 数据和表示分离

XML 的优势在于保持了用户界面和结构数据之间的分离，把数据从表示中分离出来，能够无缝集成众多来源的数据。例如，可以将用户信息、采购订单、研究结果、账单支付、医疗记录、目录数据以及其他来源转换为中间层上的 XML，以便能够像 HTML 数据一样很容易地联机交换数据。

5. 结构化和集成数据

XML 对于大型和复杂的文档是理想的，因为数据是结构化的。这不仅使用户可以定义文档中的元素词汇表，而且还可以指定元素之间的关系。例如，如果要将销售客户的地址一起放在 Web 页面上，就需要有每个客户的电话号码和电子邮件地址。如果向数据库中输入数据，除了可确保没有遗漏的字段外，还需要每部书都有一个作者，当没有数据输入时还可提供一个默认值。

XML 也提供客户端的包括机制，可以根据多种来源集成数据并将其作为一个文档来显示。数据还可以马上进行重新排列。数据的各个部分可以根据用户的操作显示或隐藏。当处理大型的信息仓库时，关系型数据库是极为有用的。

1.2 XML 和 HTML 的区别

SGML 和 HTML 是 XML 的先驱，XML 与 HTML 之间存在着很大的不同，下面将简单进行介绍。

1.2.1 认识 SGML

在介绍 XML 的发展历史时提到过 SGML。因此，XML 并不是一个新概念，它来源于 SGML，是一种比 HTML 更早的标记语言标准。

SGML 是英文 Standard Generalized Markup Language 的缩写，通常会被称为“通用标记语言标准”。它是标记语言的标准，所有的标记语言都要依照 SGML 制定。其覆盖面积非常广泛，凡是有一定格式的文件都属于 SGML，例如报告和乐谱等。HTML 是网络上最常见的一种基于 SGML 的标记语言。因此，有些地方会亲切地称呼它为 HTML 的“母亲”。

SGML 是一个庞大且复杂的系统，其应用标准包括超媒体文档标记方面的超媒体语言 HyTime、文献样式语义和规范说明语言 DSSSL、可扩展的样式语言 XSL、可扩展的链接语言 XLL 等。SGML 规模庞大、功能丰富，具有各种选项。它用来标记文献可以使文献信息不依赖于特定的软硬件，而且具有方便的可互操作性和格式的可转换功能，以适合多种应用或反复使用的目的。这些优点使它备受关注，成为文献信息加工、处理、存储和发布的主流技术之一。

1.2.2 与 HTML 的区别

HTML 虽然使用最为广泛，但是现在网络应用的快速发展，仅仅靠 HTML 单一文件类型来处理千变万化的文档和数据已经力不从心，而且 HTML 本身语法十分不严密，严重影响网络信息的传送和共享。人们早已开发探讨用什么来满足网络上各种应用的需要，使用 SGML 是可以的，但是它过于庞大，编程复杂，于是最终选择了“减肥”之后的 SGML——XML 作为下一代 Web 运用的数据传输和交互的工具。XML 是“减肥”之后的简化版，它省略其中比较复杂和不常用的部分。因此，有些地方还会称它是 HTML 的第二个“母亲”。

XML 是用来存放数据的，它不是 HTML 的替代品，而且它们是两种不同用途的语言，不同之处如下所示。

1. 定位不同

HTML 是在 SGML 定义下的一个描述性语言，只是 SGML 的应用；而 XML 是 SGML 的一个简化版本，是 SGML 的一个子集。

2. 用途不同

HTML 是被设计用于显示数据的，重点是显示数据以及如何更好地显示数据样式；而 XML 是被设计用于描述数据的，重点是什么是数据，以及如何存放数据。

3. 扩展性不同

HTML 易学、易于访问，但难以数据重用，可扩展性差；XML 比 HTML 更加容易

扩展，它表示了数据的逻辑结构，可以为不同的应用以不同方式加以分析，同时进行严格的语法和语义检查。

4. 内容不同

HTML 文档主要包含显示格式，主要是为了浏览，而不是为计算机使用，是显示格式描述语言；XML 文档将数据与显示格式分离，是数据格式描述语言，为信息开放、共享和交互提供基础。

5. 语法控制不同

HTML 的语法规则比较多元化，具有很大的灵活性。其文件结构比较松散，不能很容易地转换为其他类型格式，比较难用程序做大量而有效地处理，数据再利用的潜力大大降低。而 XML 的语法比 HTML 严格，例如，区分大小写、元素必须进行嵌套，以及属性值必须括起来等。

XML、SGML 和 HTML 都将继续用于适合的地方，它们谁也不能被另外两种语言来代替，而且它们中的任何一个都不会使另外的两个废弃。

对于像新闻、网络日记、论坛留言等大部分短期数据，HTML 仍是 Web 上快速发布数据最简单的方法；如果数据要长期使用，并且需要更多的一些结构，推荐使用 XML。与 HTML 和 XML 不同，SGML 可能不会在 Internet 上被广泛接受，因为它不是为某个网络协议而设计的，也从来没有为某个网络协议的需求而优化过。对于高端、复杂结构的发布应用，SGML 将继续使用。

1.3 XML 开发工具

XML 是一种基于文本格式的语言，可以使用记事本和写字板的文本编辑器创建，也可以使用创建简单文本文件的处理程序进行创建。开发 XML 时可以使用多种不同的开发工具。在本书中将使用 Dreamweaver 进行开发，除了该工具外，读者还可以选择以下几种工具。

- **Notepad** 这是一种最直接、最简单的文本编辑工具，它是 Windows 自带的编辑工具，在附件中就可以找到。
- **Microsoft XML Notepad** 这是一款微软专门为设计 XML 文档而提供的编辑软件，可以借助它验证 XML 文档的有效性。
- **Visual InterDev** 这是用来开发 Web 应用程序的，不仅仅局限于 XML，还可以开发 ASP、HTML 和 XSL 样式单等。
- **Microsoft XML Tree Viewer** 利用这个工具可以把 XML 文档的内容用树的结构形式显示出来。
- **Microsoft XML Validated** 使用这个工具可以检查 XML 文档是否“格式良好的”以及其有效性，且对错误发生警告。
- **Microsoft XSL Debugger** 样式表文件的复杂性使开发人员在编写时容易出现错误，这个工具就是帮助用户调试样式表文件的，这样可以把复杂枯燥的调试