

21世纪高等教育计算机规划教材



# XML 基础教程

## (第2版)

Beginning XML

■ 李淑娣 主编  
■ 赵培植 副主编

- 零点起步，重点涵盖 XML 语法、显示与应用
- 实例丰富，从实际出发深入讲解相关知识点
- 突出实践，各章上机指导培养动手开发能力



人民邮电出版社  
POSTS & TELECOM PRESS

21世纪高等教育计算机规划教材

COMPUTER

# XML 基础教程

(第2版)

Beginning XML

■ 李淑娣 主编

■ 赵培植 副主编



人民邮电出版社

北京

## 图书在版编目 (C I P) 数据

XML基础教程 / 李淑娣主编. — 2版. — 北京: 人民邮电出版社, 2013. 8  
21世纪高等教育计算机规划教材  
ISBN 978-7-115-31961-6

I. ①X… II. ①李… III. ①可扩展语言—程序设计—高等学校—教材 IV. ①TP312

中国版本图书馆CIP数据核字(2013)第132356号

## 内 容 提 要

XML 是 W3C 组织提出的一种可扩展标记语言,是独立于计算机平台的数据交换规范。本书由浅入深、循序渐进地讲述了 XML 的基本知识和基本应用,全书共分为 11 章,主要内容包括 XML 概述、XML 语法、文档类型定义 (DTD)、XML 与 CSS、可扩展样式表语言转换 (XSLT)、文档对象模型 (DOM)、XML 与数据库、XML 与正则表达式、与 Java 和 .NET 语言结合使用等,最后还给出了一个综合实例:在线成绩管理系统。书中既有对 XML 语法等基础知识的讲解,也有对 XML 操作等基本应用的讲授,书中还介绍了 XML 在数据库、Java、.NET 等领域的前沿应用。

本书可作为普通高等院校 XML 相关课程的教材,也可作为 XML 初学者和相关 XML 培训机构的参考用书。

- 
- ◆ 主 编 李淑娣  
副 主 编 赵培植  
责任编辑 刘 博  
责任印制 彭志环 杨林杰
  - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号  
邮编 100061 电子邮件 315@ptpress.com.cn  
网址 <http://www.ptpress.com.cn>  
北京艺辉印刷有限公司印刷
  - ◆ 开本: 787×1092 1/16  
印张: 18 2013 年 8 月第 2 版  
字数: 469 千字 2013 年 8 月北京第 1 次印刷
- 

定价: 39.00 元

读者服务热线: (010)67170985 印装质量热线: (010)67129223  
反盗版热线: (010)67171154

## 第2版前言

随着网络技术的飞速发展, WWW 应用已深入到千家万户。然而, 在网络信息丰富的同时, 网络数据量也是水涨船高, 如何有效管理并完美显示网页内容就成为网络发展的一项亟待解决的重要问题, XML 技术正是针对这一问题的创新性成果。

XML 技术从最初提出到广泛应用经历了很长的时间, 以 XML 技术为中心, 引伸出对 XML 的处理、查询、转换、设计等技术, 其应用已扩展到网络服务、数据库和电子商务等领域。由于篇幅所限, 本书只就 XML 语法、XML 文档的显示和应用程序对 XML 文档的使用等方面重点进行了介绍, 希望初次接触 XML 技术的读者能够抓住基本重点, 而其他方面则一笔带过或不作介绍。

在开始本书的学习之前, 读者应当具备 HTML 的基本知识以及任何一门高级编程语言(如 C/C++、Java、C#等)的使用经验。对于 XML 编程将涉及的其他技术, 本书在相关章节将会进行基础知识的介绍。

本书首先从 XML 的发展历程开始, 逐步展开介绍了 XML 的各种相关技术。全书共分 11 章, 基本包含了 XML 语法、显示和应用等方面的内容。

其中:

第 1 章简单介绍了 XML 的形成、发展与前景, 论述了学习 XML 的重要性; 本章还概述了 XML 的相关技术和应用领域, 举例说明了建立 XML 方法与过程。

第 2 章讲述了 XML 的语法知识, 只有牢固掌握了 XML 的语法知识, 才能写出符合规范的 XML 文档; 本章还简要介绍了字符集的相关内容。

第 3 章详细讲述了文档类型定义(DTD)的相关知识, DTD 的用途就在于检验写出的 XML 文档是否跟意向中的 XML 文档结构一致; 实现相同功能的还有 Schema(XML 架构); 命名空间是解决命名冲突一种方法。

第 4 章简单介绍了 CSS 的概念以及其语法结构, 并通过示例说明如何使用 CSS 与 XML 相结合, 实现数据与显示方式的分离。

第 5 章详细讲解了 XSL 的 3 个部分——XSLT、XPath、XSL-FO, 重点介绍了 XSLT 的语法结构与使用方法, 通过示例使读者对 XML 的优点有更加深入的体会。

第 6 章讲解了有关 DOM 的基本知识, 主要包括 DOM 的基本组成、DOM 接口规范中的 4 个基本接口、Microsoft 公司的 MSXML 文档对象模型的实现、关于 DOM 的一些应用等, 最后给出了一个现实应用中的实例程序。

第 7 章介绍了 XML 与数据库的基本知识, 主要包括 XML 与数据库的发展状况、XML 的数据交换机制、XML 的数据存储机制、XML 的数据源对象和 XML 的几种重要的数据交换技术等, 最后给出一个 XML 与关系数据库的简单实例程序。

第 8 章介绍了如何在 XML 中应用正则表达式, 并给出了几个简单实例。

第 9 章讲解了 XML 在 Java 中的典型应用, 其中涉及 JDOM、JAXB 等工具的用法。

第 10 章讲解了 XML 在 C# 中的典型应用, 其中涉及 XmlDocument、XmlReader、XmlWriter 等对象的使用。

第 11 章介绍了一个 XML 的综合案例——在线成绩管理系统的实现方法。

本书由李淑娣（保定职业技术学院）主编，赵培植（兰州资源环境职业技术学院）副主编，李淑娣编写了第 1 章至第 6 章，赵培植编写了第 7 章至第 11 章。

由于编者水平有限，书中难免存在不妥之处，恳请读者批评指正。

编 者  
2013 年 6 月

# 目 录

|                            |    |                                 |    |
|----------------------------|----|---------------------------------|----|
| 第 1 章 XML 概述 .....         | 1  | 2.2.4 CDATA 段 .....             | 23 |
| 1.1 什么是 XML .....          | 1  | 2.2.5 注释 .....                  | 24 |
| 1.1.1 SGML 的诞生 .....       | 1  | 2.3 XML 物理结构 .....              | 26 |
| 1.1.2 XML 是什么 .....        | 2  | 2.3.1 什么是实体 .....               | 26 |
| 1.2 为什么要学习 XML .....       | 4  | 2.3.2 实体的类型 .....               | 27 |
| 1.2.1 可重用性 .....           | 4  | 2.3.3 实体的使用 .....               | 28 |
| 1.2.2 可扩展性 .....           | 4  | 2.4 ASCII 字符集 .....             | 29 |
| 1.2.3 Web 应用 .....         | 4  | 2.4.1 ISO 字符集 .....             | 30 |
| 1.2.4 数据处理 .....           | 4  | 2.4.2 MacRoman 字符集 .....        | 31 |
| 1.3 XML 相关技术 .....         | 5  | 2.5 Unicode 字符集 .....           | 31 |
| 1.3.1 文档类型定义 .....         | 5  | 2.5.1 UTF-8 .....               | 31 |
| 1.3.2 可扩展样式语言 .....        | 5  | 2.5.2 通用字符集 .....               | 31 |
| 1.3.3 可扩展链接语言 .....        | 5  | 2.5.3 如何使用 Unicode 编写 XML ..... | 32 |
| 1.4 XML 实际应用 .....         | 6  | 小结 .....                        | 32 |
| 1.5 XML 的发展前景 .....        | 6  | 习题 .....                        | 32 |
| 1.5.1 网络服务领域 .....         | 7  | 上机指导 .....                      | 34 |
| 1.5.2 数据库领域 .....          | 7  | 实验一：元素和属性 .....                 | 34 |
| 1.5.3 电子商务领域 .....         | 7  | 实验二：CDATA 段和注释 .....            | 35 |
| 1.6 XML Spy 开发环境 .....     | 8  | 实验三：语法综合 .....                  | 35 |
| 1.6.1 XML Spy 概述 .....     | 8  | 第 3 章 文档类型定义 ( DTD ) .....      | 37 |
| 1.6.2 XML Spy 的安装 .....    | 9  | 3.1 什么是 DTD .....               | 37 |
| 1.6.3 一个 XML 文档的简单创建 ..... | 10 | 3.1.1 DTD 概述 .....              | 37 |
| 小结 .....                   | 12 | 3.1.2 第一个 DTD 示例 .....          | 38 |
| 习题 .....                   | 13 | 3.1.3 DTD 的基本结构 .....           | 40 |
| 上机指导 .....                 | 13 | 3.1.4 定义元素及其后代 .....            | 41 |
| 实验一：创建 XML 文档 .....        | 13 | 3.1.5 定义元素属性 .....              | 43 |
| 实验二：编辑 XML 文档内容 .....      | 14 | 3.1.6 DTD 中的注释 .....            | 44 |
| 实验三：简单的 XML 文档 .....       | 14 | 3.1.7 在文档间共享通用的 DTD .....       | 44 |
| 第 2 章 XML 语法 .....         | 16 | 3.2 DTD 中的属性声明 .....            | 45 |
| 2.1 什么是规范的 XML 文档 .....    | 16 | 3.2.1 在 DTD 中声明属性 .....         | 46 |
| 2.2 XML 逻辑结构 .....         | 17 | 3.2.2 声明多个属性 .....              | 46 |
| 2.2.1 XML 的整体结构 .....      | 17 | 3.2.3 指定属性的默认值 .....            | 46 |
| 2.2.2 XML 元素 .....         | 19 | 3.2.4 属性类型 .....                | 47 |
| 2.2.3 元素属性 .....           | 22 | 3.2.5 预定义属性值 .....              | 49 |

|                                                   |           |                                           |           |
|---------------------------------------------------|-----------|-------------------------------------------|-----------|
| 3.3 实体和外部 DTD 子集 .....                            | 49        | 4.2.6 ID 选择符 (ID Selectors) .....         | 78        |
| 3.3.1 内部通用实体 .....                                | 50        | 4.2.7 属性选择符<br>(Property Selectors) ..... | 78        |
| 3.3.2 外部通用实体 .....                                | 51        | 4.2.8 类选择符 (Class Selectors) .....        | 79        |
| 3.3.3 内部参数实体 .....                                | 52        | 4.2.9 其他选择方式 .....                        | 79        |
| 3.3.4 外部参数实体 .....                                | 52        | 4.3 属性 .....                              | 79        |
| 3.3.5 根据片段创建文档 .....                              | 53        | 4.3.1 字体属性 .....                          | 79        |
| 3.3.6 结构完整的文档中的实体和 DTD .....                      | 55        | 4.3.2 颜色属性 .....                          | 82        |
| 3.4 Schema 简介 .....                               | 56        | 4.3.3 背景属性 .....                          | 83        |
| 3.4.1 Schema 概述 .....                             | 56        | 4.3.4 文本属性 .....                          | 85        |
| 3.4.2 定义元素及其后代 .....                              | 58        | 4.3.5 框属性 .....                           | 85        |
| 3.4.3 Schema 的应用 .....                            | 60        | 4.4 CSS 的书写规范 .....                       | 88        |
| 3.5 XML 命名空间 .....                                | 63        | 4.5 XML 与 CSS 的综合运用 .....                 | 89        |
| 3.5.1 什么是命名冲突 .....                               | 63        | 小结 .....                                  | 92        |
| 3.5.2 解决命名冲突途径 .....                              | 63        | 习题 .....                                  | 93        |
| 3.5.3 命名空间的使用 .....                               | 64        | 上机指导 .....                                | 93        |
| 3.5.4 DTD 与命名空间 .....                             | 64        | 实验一: 美化导航条 .....                          | 93        |
| 小结 .....                                          | 65        | 实验二: 字体属性设置 .....                         | 94        |
| 习题 .....                                          | 65        | 实验三: XML 与 CSS 综合设置 .....                 | 94        |
| 上机指导 .....                                        | 67        | <b>第 5 章 可扩展样式表语言<br/>转换 (XSLT) .....</b> | <b>96</b> |
| 实验一: 练习使用 XMLSpy 自动<br>生成 DTD 文档 .....            | 68        | 5.1 什么是 XSL .....                         | 96        |
| 实验二: 练习使用 XMLSpy 的 Grid<br>模式编辑 DTD 文档 .....      | 68        | 5.1.1 XSL 构成 .....                        | 96        |
| 实验三: DTD 综合 .....                                 | 69        | 5.1.2 树形结构 .....                          | 97        |
| <b>第 4 章 XML 与 CSS .....</b>                      | <b>71</b> | 5.1.3 XSL 样式单文档 .....                     | 97        |
| 4.1 什么是 CSS .....                                 | 71        | 5.1.4 在何处进行 XML 变换 .....                  | 98        |
| 4.1.1 CSS 的历史 .....                               | 71        | 5.2 创建一个 XSL 实例 .....                     | 99        |
| 4.1.2 CSS 的编写环境以及功能简要<br>说明 .....                 | 72        | 5.2.1 源代码及显示效果 .....                      | 99        |
| 4.1.3 CSS 的使用方式 .....                             | 73        | 5.2.2 各部分详解 .....                         | 101       |
| 4.2 选择元素 .....                                    | 76        | 5.3 XSL 模板 .....                          | 103       |
| 4.2.1 类型选择符 (Type Selectors) .....                | 77        | 5.3.1 模板的简单应用 .....                       | 103       |
| 4.2.2 通配选择符<br>(Universal Selectors) .....        | 77        | 5.3.2 xsl:apply-templates 元素 .....        | 104       |
| 4.2.3 包含选择符<br>(Descendant Selectors) .....       | 77        | 5.3.3 select 特性 .....                     | 104       |
| 4.2.4 子对象选择符 (Child Selectors) .....              | 77        | 5.3.4 默认的模板规则 .....                       | 108       |
| 4.2.5 相邻选择符<br>(Adjacent Sibling Selectors) ..... | 78        | 5.4 XSL 元素 .....                          | 108       |
|                                                   |           | 5.4.1 XSL 元素构成 .....                      | 108       |
|                                                   |           | 5.4.2 循环 xsl:for-each .....               | 111       |
|                                                   |           | 5.4.3 排序 xsl:sort .....                   | 112       |
|                                                   |           | 5.4.4 选择 xsl:if 和 xsl:choose .....        | 112       |

|                              |     |                                           |     |
|------------------------------|-----|-------------------------------------------|-----|
| 5.4.5 xsl:fallback 元素 .....  | 114 | 第 6 章 文档对象模型 (DOM) .....                  | 138 |
| 5.4.6 XSL 函数集 .....          | 115 | 6.1 DOM 的组成 .....                         | 138 |
| 5.5 匹配节点的模式 .....            | 118 | 6.1.1 一棵简单的 DOM 树 .....                   | 138 |
| 5.5.1 匹配根节点 .....            | 118 | 6.1.2 DOM 的核心部分 .....                     | 140 |
| 5.5.2 匹配元素名 .....            | 118 | 6.1.3 DOM 接口规范中的 4 个<br>基本接口 .....        | 142 |
| 5.5.3 使用 “/” 字符匹配子节点 .....   | 118 | 6.2 DOM 的接口 .....                         | 146 |
| 5.5.4 使用 “//” 字符匹配子节点 .....  | 119 | 6.2.1 为什么要使用 DOM 接口 .....                 | 146 |
| 5.5.5 通过 ID 匹配 .....         | 119 | 6.2.2 接口与实现 .....                         | 147 |
| 5.5.6 使用 @ 来匹配特性 .....       | 120 | 6.2.3 MSXML 文档对象模型的接口<br>一览及重要接口介绍 .....  | 148 |
| 5.5.7 使用 comments() 注释 ..... | 120 | 6.3 DOM 的应用 .....                         | 162 |
| 5.5.8 使用 pi() 来匹配处理指令 .....  | 121 | 6.3.1 添加 DOM 处理引用 .....                   | 162 |
| 5.5.9 用 text() 来匹配文本节点 ..... | 121 | 6.3.2 加载 XML 文档 .....                     | 164 |
| 5.5.10 使用 “或” 操作符 .....      | 122 | 6.3.3 处理节点 .....                          | 166 |
| 5.6 输出格式与编码问题 .....          | 122 | 6.3.4 保存文档对象 .....                        | 171 |
| 5.6.1 输出文档 .....             | 122 | 6.3.5 验证文档 .....                          | 173 |
| 5.6.2 输出文本 .....             | 123 | 6.3.6 一个实例程序 .....                        | 174 |
| 5.6.3 输出元素 .....             | 123 | 小结 .....                                  | 176 |
| 5.6.4 输出属性 .....             | 124 | 习题 .....                                  | 177 |
| 5.6.5 输出指令 .....             | 124 | 上机指导 .....                                | 178 |
| 5.6.6 输出注释 .....             | 124 | 实验一: 利用 DOM 加载指定内容的<br>XML 文档片段 .....     | 178 |
| 5.6.7 输出消息 .....             | 124 | 实验二: 利用 DOM 修改 XML 文档中<br>指定节点的属性信息 ..... | 179 |
| 5.6.8 替换名称空间 .....           | 125 | 实验三: 利用 DOM 在 XML 文档中<br>删除一个元素节点 .....   | 180 |
| 5.6.9 空白符的输出 .....           | 125 | 第 7 章 XML 与数据库 .....                      | 183 |
| 5.7 格式对象 FO .....            | 125 | 7.1 XML 技术与数据库发展 .....                    | 183 |
| 5.7.1 XSL-FO 文档 .....        | 125 | 7.1.1 数据库技术的发展 .....                      | 183 |
| 5.7.2 XSL-FO 区域 .....        | 126 | 7.1.2 XML 与数据库技术的结合 .....                 | 184 |
| 5.7.3 XSL-FO 输出 .....        | 127 | 7.2 XML 的数据交换与存储机制 .....                  | 187 |
| 5.7.4 XSL-FO FLOW .....      | 127 | 7.2.1 XML 的数据交换机制 .....                   | 187 |
| 5.7.5 XSL-FO 页面 .....        | 127 | 7.2.2 XML 的数据存取机制 .....                   | 189 |
| 5.7.6 XSL-FO 块状区域 .....      | 128 | 7.3 XML 数据源对象 .....                       | 191 |
| 5.7.7 XSL-FO 列表 .....        | 131 | 7.4 XML 数据交换技术 .....                      | 191 |
| 5.7.8 XSL-FO 表格 .....        | 131 | 7.4.1 ADO 控件技术 .....                      | 192 |
| 5.7.9 XSL-FO 参考资料 .....      | 132 | 7.4.2 HTTPXML 对象技术 .....                  | 193 |
| 小结 .....                     | 133 | 7.4.3 ODBC2XML 转换工具 .....                 | 195 |
| 习题 .....                     | 134 |                                           |     |
| 上机指导 .....                   | 134 |                                           |     |
| 实验一: 图书信息示例 .....            | 134 |                                           |     |
| 实验二: 模板的运用和设置 .....          | 135 |                                           |     |
| 实验三: XSLT 设置显示样式 .....       | 136 |                                           |     |

|                                              |     |                                     |     |
|----------------------------------------------|-----|-------------------------------------|-----|
| 7.4.4 XOSL 转换工具                              | 195 | 9.2.2 删除和修改节点                       | 222 |
| 7.4.5 WDDX Web 分布式数据交换                       | 197 | 9.3 用 JAXB 解析 XML                   | 224 |
| 7.5 一个简单的 XML 与数据库的应用                        | 198 | 9.3.1 下载与安装 JAXB                    | 224 |
| 小结                                           | 201 | 9.3.2 XJC 简介                        | 225 |
| 习题                                           | 202 | 9.3.3 JXL 简介                        | 225 |
| 上机指导                                         | 202 | 9.3.4 查看用来映射的 XML Schema 文档         | 225 |
| 实验一: 使用 SQL Server 2000 创建数据库                | 202 | 9.4 项目开发                            | 226 |
| 实验二: 使用 ADO 操作 SQL Server 2000 数据库并生成 XML 文档 | 204 | 9.4.1 创建项目                          | 227 |
| 实验三: 使用 ADO 操作数据库并利用 DOM 生成 XML 文档           | 205 | 9.4.2 利用 XJC 生成 Java 类              | 227 |
| <b>第 8 章 XML 与正则表达式</b>                      | 208 | 9.4.3 存储了字典表的 Excel 文档              | 230 |
| 8.1 正则表达式在 XML 中的应用                          | 208 | 小结                                  | 233 |
| 8.1.1 在 XML Schema 中的应用                      | 208 | 习题                                  | 233 |
| 8.1.2 在 XPath 2.0 中的应用                       | 208 | 上机指导                                | 233 |
| 8.1.3 在 XSLT 2.0 中的应用                        | 209 | 实验一: DOM 解析 XML                     | 233 |
| 8.2 XML 正则表达式简介                              | 209 | 实验二: SAX 解析 XML                     | 235 |
| 8.2.1 元字符和普通字符                               | 209 | 实验三: DOM4J 解析 XML                   | 237 |
| 8.2.2 量词                                     | 209 | <b>第 10 章 XML 在 C# 中的典型应用</b>       | 240 |
| 8.2.3 字符转义与字符类                               | 210 | 10.1 C# 中的 XML DOM                  | 240 |
| 8.2.4 字符组的使用                                 | 211 | 10.1.1 XML DOM 的操作对象 XmlDocument    | 240 |
| 8.2.5 正则表达式分支                                | 211 | 10.1.2 使用 XML 文件分析 XmlDocument 中的对象 | 240 |
| 小结                                           | 211 | 10.1.3 使用 DOM 对象获取 XML 文件           | 241 |
| 习题                                           | 211 | 10.1.4 使用 DOM 对象获取 XML 文件中的指定节点     | 242 |
| 上机指导                                         | 212 | 10.1.5 使用 DOM 对象改变 XML 文件的数据顺序      | 244 |
| 实验一: 使用正则表达式获取指定元素所有属性的集合                    | 212 | 10.2 XML 文件读取器——XmlReader           | 245 |
| 实验二: 采用 JS 正则表达式验证 XML 文件结构                  | 213 | 10.2.1 XmlReader 的作用                | 245 |
| 实验三: JS 正则表达式判断是否为数字                         | 215 | 10.2.2 对 XML 的验证                    | 246 |
| <b>第 9 章 XML 在 Java 中的典型应用</b>               | 217 | 10.2.3 使用 XmlReader 读取 XML 文件的一部分   | 246 |
| 9.1 用 JDOM 解析 XML 文档                         | 217 | 10.2.4 使用 XmlTextReader 读取整个 XML 文件 | 248 |
| 9.1.1 准备工作                                   | 217 | 10.3 XML 文件编写器——XmlWriter           | 250 |
| 9.1.2 创建 Java 类                              | 218 | 10.3.1 XmlWriter 的作用                | 250 |
| 9.2 用 JDOM 处理 XML 文档                         | 220 | 10.3.2 XmlWriter 对 XML 文件的验证        | 250 |
| 9.2.1 创建 XML 文档                              | 220 |                                     |     |

|                                        |            |                                   |     |
|----------------------------------------|------------|-----------------------------------|-----|
| 10.3.3 用 XmlWriter 创建并编辑 XML 文件 .....  | 250        | 11.1.1 系统功能简介 .....               | 263 |
| 10.4 XML 与 DataSet 的交互 .....           | 252        | 11.1.2 系统架构 .....                 | 263 |
| 10.4.1 将 XML 文件转化为 DataSet 数据集 .....   | 252        | 11.2 学生信息管理模块 .....               | 264 |
| 10.4.2 将 DataSet 数据集转换为 XML 文件 .....   | 254        | 11.2.1 XML 结构 .....               | 264 |
| 小结 .....                               | 257        | 11.2.2 学生信息模型 .....               | 265 |
| 习题 .....                               | 257        | 11.2.3 访问学生信息 DAO .....           | 266 |
| 上机指导 .....                             | 257        | 11.2.4 访问学生信息 DAO 实现类 .....       | 267 |
| 实验一: XmlDocument 对象操作                  |            | 11.2.5 StudentDAOImpl 单元测试类 ..... | 269 |
| XML 文件 .....                           | 257        | 11.2.6 XML 工具类 .....              | 271 |
| 实验二: XPath 查询 XML 内容 .....             | 260        | 11.3 学生成绩管理模块 .....               | 271 |
| 实验三: LINQ to XML 操作 XML .....          | 261        | 11.3.1 XML 结构 .....               | 271 |
| <b>第 11 章 综合案例——XML 在线成绩管理系统 .....</b> | <b>263</b> | 11.3.2 学生成绩模型 .....               | 272 |
| 11.1 系统功能简介和架构设计 .....                 | 263        | 11.3.3 访问学生成绩 DAO .....           | 274 |
|                                        |            | 11.3.4 访问学生成绩 DAO 实现类 .....       | 274 |
|                                        |            | 11.3.5 GradeDAOImpl 单元测试类 .....   | 277 |
|                                        |            | 小结 .....                          | 278 |

# 第 1 章

## XML 概述

XML 是 W3C（万维网联盟）提出的一种可扩展标记语言，其全称是 eXtensible Markup Language，它是随着人们对信息传输要求的不断提高而产生的一种新技术。通过本章的学习，读者将会了解到 XML 技术的具体含义及其广阔的应用前景。此外，本章将会告诉读者如何创建一个基本的 XML 文档。

### 1.1 什么是 XML

XML 是在 SGML 的基础之上发展起来的，XML 是 SGML 系列中的一种，人们熟知的 HTML 也是 SGML 家族中的一员。

#### 1.1.1 SGML 的诞生

SGML（Standard Generalized Markup Language）即标准通用标记语言。SGML 的思想最初是在 IBM 的一个信息管理项目中产生的，称为 GML（通用标记语言），是一种 IBM 格式化文档语言，用于对文档组织结构、各部件及其之间的关系进行描述。由于在当时的信息交换过程中，经常会发生数据格式不同的问题，随着网络技术的不断发展，这一问题日益严重，制约了人们的信息交流。1986 年，国际标准化组织（ISO）采纳了 IBM 的这一思想，并整理为 SGML。SGML 是基于文档标记语言的一种元语言，它不仅具有良好的扩展性，而且可移植性强，在任何一种环境下都可以正常使用。

##### 1. 标记语言

标记是指一系列特殊的字符或符号，用户可以向其中插入文本来存储文档内容。标记语言（Markup Language，ML）是指通过一系列具有特定含义的符号标记，按照一定规则插入到电子文档中，以方便电子文档的使用和管理。标记语言的作用和标点符号类似，最初出现在印刷业中。它们都属于元数据的范畴，即不能单独存在，都是对文档内容及格式的说明数据。但是标记语言的结构更为复杂，功能也更为强大。

标记语言的种类很多，但是都遵循同样的原则。为了方便管理和使用文档对象，所采用的标记应该能够很容易和文档内容进行区分，易于识别。目前，在各个领域，都有专业标记语言的存在。

##### 2. SGML 文档

SGML 通过 SGML 文档的形式来表现。SGML 文档定义独立于应用平台和所使用的文本文档的格式、索引和链接信息。它为用户提供一种类似于语法的机制，用来定义文档的结构和指示文

档结构的标签。SGML 文档由 3 部分组成,即语法定义、文档类型定义和文档实例。

- (1) 语法定义:定义文档类型和文档实例的语法结构。
- (2) 文档类型定义:定义文档实例的结构和组成结构的元素类型。
- (3) 文档实例:是 SGML 文档的主体部分。

SGML 文档的结构相当严谨,其中文档类型定义(DTD)是它的核心所在。DTD 为组织文档的文档元素提供了一个框架,同时为文档元素之间的相互关系制定了规则。SGML 文档具有极强的完整性和稳定性,其可适用的范围也相当广。然而,提供如此完整和稳定功能的文档语言,导致其自身也相当复杂,难以让人掌握,因此,在 SGML 的基础之上,产生了一种新的语言,即 XML。

### 1.1.2 XML 是什么

XML 就是可扩展标记语言。标记是能够被不同计算机所理解的符号,计算机之间可通过标记处理包含各种信息的文档。XML 也诞生于出版界,由于人们看到了 SGML 的不足,因而产生了精简 SGML 的需要。简单地说,XML 就是 SGML 的一个子集,仅仅去掉了 SGML 中不经常使用的和不适应于 Web 应用的部分。无论是 XML 还是 SGML,其管理和使用方法十分简单,仅仅利用记事本程序,就可以轻松地建立一个简单的 XML 文档。例如,可以通过以下步骤创建一个 XML 文档。

- (1) 打开记事本(或任何纯文本编辑器,如 EditPlus、UltraEdit 等)。

- (2) 在记事本中输入以下内容,如图 1.1 所示。

```
<?xml version="1.0" encoding="gb2312" standalone="yes"?> <!--xml 声明-->
<红楼梦>                                     <!--根元素开始标签-->
  <别名>石头记</别名>                         <!--元素及其内容声明-->
  <作者>                                       <!--元素开始标签-->
    <名>沾</名>                               <!--元素及其内容声明-->
    <字>梦阮</字>
    <号>雪芹</号>
  </作者>                                     <!--元素结束标签-->
  <写作时间>清初</写作时间>                 <!--元素及其内容声明-->
  <故事简介>                                  <!--元素开始标签-->
    <主要内容>以贾、史、王、薛四大家族为背景,以贾宝玉、林黛玉的爱情悲剧为主要线索,描写了贾家荣、宁两府由盛而衰的过程。</主要内容>
    <主要人物>贾宝玉、林黛玉、薛宝钗、史湘云、贾迎春、贾探春、贾惜春、妙玉、晴雯、王熙凤、袭人、香菱等</主要人物>
  </故事简介>                                <!--元素结束标签-->
</红楼梦>                                    <!--根元素结束标签-->
```



注意

在 XML 文档中,要严格区分大小写。第 1 行代码全是小写。这一行的字符组合,称为 XML 声明(Declaration),详情请参见第 2.1 节。在书写 XML 代码的过程中,还要留意使用正确的标点,代码中的“<”、“>”、“?”、“=”、“”、“/”和“.”分别是英文标点中的小于号、大于号、问号、等于号、引号、左斜杠和句号。输入上述符号时,建议关闭中文输入法。

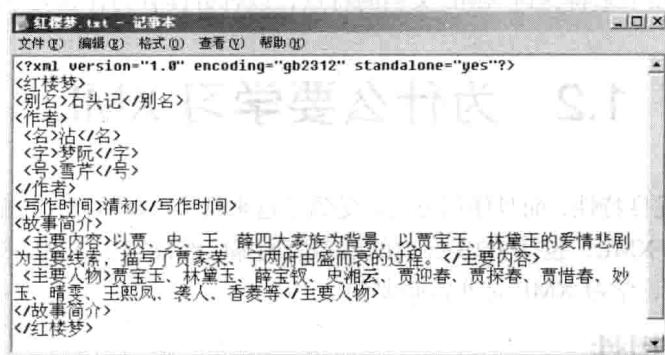


图 1.1 用记事本创建第一个 XML 文档

(3) 将文档保存为带有“.xml”后缀的文件，如“红楼梦.xml”。



注意

当使用记事本时，在“另存为”对话框中的“保存类型”下拉列表中，必须选择“所有文件”，而不是默认的“文本文件 (\*.txt)”，否则只能得到后缀为“.txt”的文本文件。

(4) 将文件保存后，本节的 XML 文档就创建成功了。

(5) 可以使用 Internet Explorer (IE) 打开并查看该文件的内容，如图 1.2 所示。

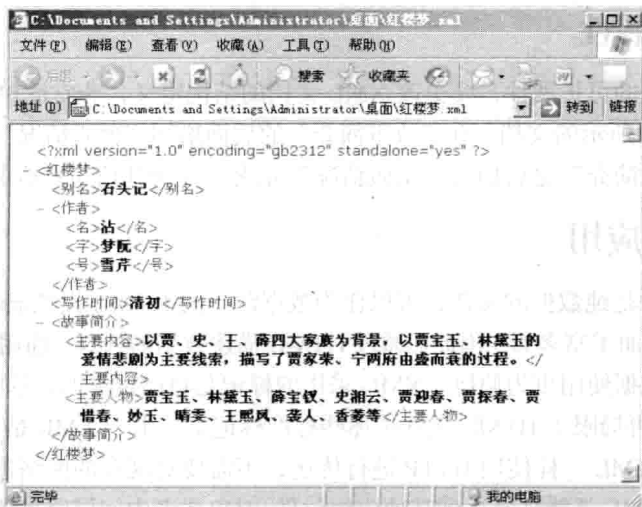


图 1.2 在 IE 上可以浏览 XML 文档

该 XML 文档用一种“尖括号标签”的标记方式，说明了有关《红楼梦》的部分信息，具体内容如下：

- (1) 《红楼梦》又叫《石头记》；
- (2) 作者曹雪芹的相关信息；
- (3) 《红楼梦》的写作时间；
- (4) 该书的主要内容和主要人物。

创建 XML 文档的过程中，除了大小写和标点符号稍微麻烦以外，不会有其他问题。在创建

了 XML 文档之后, 接下来将探讨 XML 文档的特点, 以告诉读者为什么要学习 XML。

## 1.2 为什么要学习 XML

XML 具有许多优良特性, 而且使用方便, 受到了越来越多人的青睐。目前, 许多大公司和开发人员已经开始使用 XML, 包括 B2B 在内的很多应用都已经证实了 XML 将会改变今后创建应用程序的方式, 因此, 学习 XML 是非常必要的。

### 1.2.1 可重用性

从 1.1.2 小节的示例可以看到, XML 文档是一系列的数据。这与 HTML 有很大的区别, HTML 标记与表现形式捆绑在一起, 如 “<H1>” 中的内容被显示成一号标题, “<TABLE>” 中的内容被显示成表格。而 XML 的标记和表现形式并没有任何必然的关联。表现 XML 形式的职责, 通常并不在 XML 文档上面。XML 文档这种内容和形式分离的设计思想, 初看起来不太友好, 实际应用起来, 却大有好处。

由于 XML 文档本身不受表现形式的羁绊, 只要对 XML 文档作适当的转换, 就可以将其变成不同的形式, 如网页、PDF 文档、Word 文档等, 达到 “一次编写, 多处使用” 的目的, 提高了内容的可重用性。

### 1.2.2 可扩展性

XML 具有非常好的可扩展性, 要更改 XML 文档的结构定义, 非常简单。

例如, 修改图 1.1 所示的文档, 在 “故事简介” 的后面增加 “出版情况”。操作方法为: 打开 XML 文档, 在 “故事简介” 之后插入 “出版情况” 元素, 并为其填写对应的内容即可。

### 1.2.3 Web 应用

XML 元数据文件是纯数据的文件, 可以作为数据源, 向 HTML 提供显示的内容。显示样式可以随 HTML 的变化而丰富多彩, 因此, 通过 HTML 描述数据的外观, 而用 XML 来描述数据本身, 将使 Web 上的数据使用更为便捷。XML 采用的标记是自己定义的, 这样数据文件的可读性就能大大提高, 也不再局限于 HTML 文件中那些标准标记了。由于 XML 是一个开放的基于文本的格式, 它可以和 HTML 一样使用 HTTP 进行传送, 不需要对现存的网络进行改变。

数据一旦建立, XML 就能被发送到其他应用软件、对象或者中间层服务器中做进一步的处理, 或者可以被发送到桌面用浏览器浏览。XML 和 HTML、脚本、公共对象模式一起为灵活的 3 层 Web 应用软件的开发提供了所需的技术。

### 1.2.4 数据处理

XML 是以文本形式来描述的一种文件格式。使用标记描述数据, 可以具体指出开始元素和结束元素, 在开始元素和结束元素之间是要表现的元素数据。这种用元素表现数据的方法可以嵌套, 因而可以表现层状或树状的数据集合。XML 作为数据库, 既具有关系型数据库 (二维表) 的特点, 也具有层状数据库 (分层树状) 的特点, 能够更好地反映现实中的数据结构。XML 还可以很方便地与数据库中的表进行相互转换。XML 是不同数据结构体的文本化描述语言, 它可

以描述线性表、树、图形等数据结构，也能描述文件化的外部数据结构，因此是一种通用的数据结构。

XML 使计算机能够很简易地存储和读取资料，并确保数据结构精确。由于 XML 是以文本形式描述的，所以适合于各种平台环境的数据交换。同样，由于使用文本来描述内容，可以越过不同平台的障碍进行正常的数据交换。当然，文本形式也会因为文字代码的不同造成不能阅读的问题，但在这一点上，XML 有着非常完美的解决方案，避免了一般语言设计的缺漏，可支持国际化及地区化的格式。

## 1.3 XML 相关技术

XML 涉及很多相关的技术，只有将这些技术结合起来，才能充分发挥 XML 的强大功能。这些技术主要包括：DTD（文档类型定义）、XSL（可扩展样式语言）、XLL（可扩展链接语言）、DOM（文档对象模型）、Namespaces（XML 命名空间）、XHTML（可扩展 HTML）等。下面将对其中比较关键的几种技术进行简单介绍。

### 1.3.1 文档类型定义

文档类型定义（DTD）是用于描述、约束 XML 文档结构的一种方法，规定了文档的逻辑结构。它可以定义文档的语法，而文档的语法反过来能够让 XML 语法分析程序确认某张页面标记使用的合法性。DTD 定义页面的元素、元素的属性以及元素和属性之间的关系，如 DTD 能够规定某个表项只能在某个列表中使用。

理想的定义应面向描述与应用程序有关的数据结构而不是如何显示数据。换句话说，应把一个元素定义为一个标题行，然后让样式表和脚本定义如何显示标题行。

DTD 不是强制性的。对于简单应用程序来说，开发人员不需要建立他们自己的 DTD，可以使用预先定义的公共 DTD，或者根本就不使用。即使某个文档已经有了 DTD，只要文档是结构完整的，语法分析程序也可以不对照 DTD 来检验文档的合法性。服务器可能已经执行了检查，所以检验的时间和带宽将得以节省。

### 1.3.2 可扩展样式语言

可扩展样式语言（XSL）是用于规定 XML 文档样式的语言。XSL 能使 Web 浏览器改变文档的表示法，如使数据的显示顺序改变，而不需要与服务器进行交互通信。通过变换样式表，同一个文档可以显示得更大，或者经过折叠只显示外面的一层，或者变为打印格式。可以设想一个适合用户学习特点的技术手册，它为初学者和更高一级的用户提供不同的样式，而且所有的样式都是根据同样的文本产生的。

XSL 凭借其可扩展性能够控制无穷无尽的标记，而控制每个标记的方式也是无穷尽的。这就给 Web 提供了高级的布局特性，如旋转的文本、多列和独立区域等。它支持国际书写格式，可以在一页上混合使用从左至右、从右至左和从上至下的书写格式。

### 1.3.3 可扩展链接语言

可扩展链接语言（eXtensible Linking Language, XLL）是一种链接语言，它支持目前 Web 上

已有的简单链接,并且将进一步扩展链接,包括结束死链接的间接链接以及可以从服务器中仅查询某个元素的相关部分的链接等。超文本标记语言 (HTML) 只执行与超文本系统概念相关的少数连接功能,只支持最简单的链接形式,这与 XML 相比有很大的差别。在为 XML 所设想的真正的超文本系统中,所有典型的超文本链接机制都将得到支持,包括以下几种类型:

- (1) 与位置无关的命名;
- (2) 双向链接;
- (3) 可以在文档外规定和管理的链接;
- (4) 元超链接 (如环路、多个窗口);
- (5) 集合链接 (多来源);
- (6) Transclusion (链接目标文档是链接源文档的一部分);
- (7) 链接属性 (链接类型)。

这些类型都可以通过 XLL 来实现。由于 XML 以 SGML 为基础,所以 XLL 基本上是 Hytime (超媒体/基于时间的结构语言, ISO10744) 的一个子集。它还遵循文本编码倡议 (TextEncoding Initiative) 规定的链接概念。

## 1.4 XML 实际应用

XML 在实际使用的过程中发挥着巨大的作用。目前,越来越多的行业开始采用 XML 来实现需要的特定功能。XML 最主要的应用主要体现在以下几个方面。

### 1. 数据交换

对于一个应用程序来说,数据交换是最基本的任务。XML 使用自定义标记存储数据信息,而且存储了各标记之间的关系,如父子关系、兄弟关系等。这使得平面文件必须要使用额外数据来存储的信息,可以隐含地保存在 XML 文档的自身结构中。

### 2. 跨平台应用开发

XML 文档不依赖于任何开发语言,各种开发语言都已经实现了与 XML 的沟通。例如,通过 XML 文档的中间介质作用,可以实现 Java 开发与 C# 开发的良好交互。

### 3. 数据库

XML 文档完全可以作为小型数据库来使用,这样就避免了少量信息必须存储到专业数据库的麻烦。当然,在数据量非常大时,使用 XML 文档来存储数据的成本是非常高的。

### 4. 配置文件

使用 XML 作为程序配置,具有与面向对象数据结构类似、轻便灵活、容易调试等优点。Java 平台和 .NET 平台均大量地使用了 XML 作为程序的配置文件,并有很多相关的类,支持读写 XML 配置文件。在这些平台上开发应用程序,掌握 XML 方面的知识无疑是很有好处的。

## 1.5 XML 的发展前景

XML 是一个新兴的网络信息描述、组织和显示语言,是 Internet 的“世界语”。它的开放性、严谨性、灵活性和结构性倍受网络开发者的青睐。Web 的飞速发展给予了 XML 充分展示自我的

空间,它提供给使用者更为强大的功能,带给程序员更为便利的开发环境。在以下领域,XML 将一展风采。

### 1.5.1 网络服务领域

网络服务被称为“IT 产业下一波浪潮”。为了争夺这一领域的主导权,Sun 公司与 Microsoft 公司正在进行激烈的争夺。然而,无论是 Sun 公司还是 Microsoft 公司,在为自己的网络服务计划宣传时,都一定会把 XML 作为一项重要内容。XML 将成为未来互联网领域占主导地位的标准通信协议,今后各类手持设备、台式计算机等产品都将安装使用 XML 可扩展标记语言。

Microsoft 公司基础的网络应用平台 .NET 体系完全是构架在 XML 之上的。在该体系中,所有中间传输的文件都以 XML 的形式传输,XML 成为 .NET 体系的血液。同时,.NET 技术的普及也带动了 XML 技术的应用。

XML 有利于信息的表达和结构化组织,从而使数据搜索更有效。XML 可以使用 URL 别名使 Web 的维护更方便,也使 Web 的应用更稳定,XML 还可以使用数字签名,使 Web 的应用更广阔。而 XML 的广泛使用必然能推动 Web 不断发展,从而开创 Web 应用的新时代。

信息发布在企业的竞争发展中起着重要的作用。服务器只需发出一份 XML 文档,客户就可根据自己的需求选择和制作不同的应用程序以处理数据。加上 XSL (eXtensible Stylesheet Language) 的帮助,使广泛的、通用的分布式计算成为可能。

### 1.5.2 数据库领域

关系型数据库行业的三大世家——IBM 公司、Oracle 公司和 Microsoft 公司都分别在它们的数据库产品中提供了对 XML 的支持。作为 Microsoft 公司 .NET 战略重要部分的 Yukon 正是基于 XML 技术的。Oracle 公司也已经推出与 XML 有关的产品 XDB (XML 数据库支持),而 IBM 公司也已实现了 DB2 和 XML Extender 的完美结合。

XML 文档可以定义数据结构,代替数据字典,用程序输出建库脚本。应用“元数据模型”技术,对数据源中不同格式的文档数据,可按照预先定义的 XML 模板,以格式说明文档结构统一描述,并提取数据或做进一步处理,最后转换为 XML 格式输出。XML—数据库—网页或文档中的表格,这三者可以互相转换。

XML 文档从本质来看就是数据库,它是数据的集合,每个文件都含有某种类型的数据。在许多方面看起来它和其他文件没什么区别,但作为一种“数据库”格式,XML 有一些优点,如它是自描述的(所用的标记描述了数据的结构和类型,尽管缺乏语义),可交换的,能够以树状或图形结构描述数据;同样它也有缺点,如它显得有些烦琐,由于要对它进行解析和文本转换,所以数据访问速度较慢。

### 1.5.3 电子商务领域

电子商务是在 20 世纪 90 年代初随着 Internet 的普及出现的。Internet 的全球性扩大了交易范围,电子信息的传递降低了交易的成本,数据加密、电子认证、安全电子交易等一系列措施,提高了交易的安全性。但是,在实现跨平台跨系统的数据交换方面还不是很方便,仍需要在多种文件格式间进行转换。

XML 的出现为电子商务注入了新的活力,需要做的第一步就是将企业之间日常交流和交换的信息尽可能地电子化、统一化,来满足不同商业系统之间的数据交换需求。Microsoft 公司的电子