网页设计与
Web 数据库发布技术

魏应彬　周星　康耀红　编著
符华儿　审

清 华 大 学 出 版 社

需要全本请访问 www.ertongbook.com

(京)新登字 158 号

内 容 简 介

网页设计是基于互联网的各种应用的基础,Web 数据库发布技术是动态网页制作的核心。本书简明扼要地介绍了网页设计中的 HTML 语言、FrontPage 2000 网页制作工具和 XML 语言,以 Web 数据库发布为主线,全面系统地介绍了当今实现动态网页的流行技术——ASP、PHP、JSP 和 ColdFusion 技术。

书中给出了大量经过调试运行的实例,将理论阐述融于具体例题之中,方便读者学习和理解。本书所有例题的源程序都附在配书光盘中。

本书可作为高等院校相关专业的教学用书,也可作为从事网站开发人员的自学或培训教材。

版权所有,翻印必究。
本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

书　　名: 网页设计与 Web 数据库发布技术
作　　者: 魏应彬　周星　康耀红
出 版 者: 清华大学出版社(北京清华大学学研大厦,邮编 100084)
　　http:// www .tup .tsinghua .edu .cn
印 刷 者: 北京密云胶印厂
责任编辑: 王敏稚(E-mail: wangmz@tup .tsinghua .edu .cn)
发 行 者: 新华书店总店北京发行所
开　　本: 787×1092 1/ 16　印张: 24　字数: 548 千字
版　　次: 2002 年 1 月第 1 版　2002 年 1 月第 1 次印刷
书　　号: ISBN 7-900637-79-6
印　　数: 0001～5000

定　　价: 27 .00 元(含盘)



互联网的蓬勃发展对人类生活的各个方面产生了深刻影响，Web 网页是互联网上信息组织的基本形式。在学习网页制作的过程中，除了掌握 HTML 语言和如何用网页制作工具（如 FrontPage 2000、Dreamweaver）来进行网页设计之外，更重要的是要对网页设计中的核心内容，即 Web 数据库发布技术熟练运用。只有这样，才能正确理解网站上的网上购物、网上聊天、网上考试、网上交友等功能的实现原理，开发出实用的、动态的网页。

本书全面整合了 Web 数据库发布技术、网页制作技术和软件开发理论，形成了一个有机的整体。力图使读者通过本书的学习，全面系统地掌握网页设计和 Web 数据库发布的技术，具有独立开发一个 Web 站点的能力。

本书在编写时本着循序渐进、由浅入深、易学易懂、实用性强的原则，在总结多年教学与开发经验的基础上，突出重点、难点，避免晦涩难懂的理论描述，通过大量的例题来说明需掌握的理论知识。

本书的第 4、5、6、7 章分别介绍了实现 Web 数据库发布的四种流行技术。在学习本书的过程中，可根据实际情况有选择地学习。

本书由魏应彬、周星、康耀红编著，周星统稿。第 1 章由魏应彬、康耀红编写；第 2 章由魏应彬、李朝晖、黄海编写；第 3 章由云敏编写；第 4 章、第 5 章由周星编写；第 6 章、第 7 章由魏应彬编写；第 8 章第 1 节、第 2 节、第 5 节由周星、康耀红编写；第 8 章第 3 节、第 4 节由魏应彬、林天编写。

在本书的出版过程中，得到了清华大学出版社的大力帮助，在此深表谢意。

由于作者水平有限，书中的不妥之处，请广大读者批评指正。

作 者

2001 年 10 月

第 1 章　Web 技术 …… 1
1.1　Web 网页 …… 1
1.1.1　什么是 Web …… 1
1.1.2　静态网页 …… 1
1.1.3　动态网页 …… 2
1.1.4　浏览器/ 服务器模式 …… 2
1.2　Web 数据库 …… 3
1.2.1　什么是 Web 数据库 …… 3
1.2.2　Web 数据库访问原理 …… 3
1.2.3　Web 数据库访问技术 …… 4
1.3　使用 SQL 语言 …… 5
1.3.1　创建表 …… 5
1.3.2　数据查询 …… 5
1.3.3　插入记录 …… 6
1.3.4　删除记录 …… 6
1.3.5　更新记录 …… 7

第 2 章　网页设计技术 …… 8
2.1　HTML 语言入门 …… 8
2.1.1　一个简单的例子 …… 8
2.1.2　HTML 基本格式 …… 8
2.2　超链接 …… 9
2.2.1　URL …… 9
2.2.2　超链接的基本格式 …… 9
2.2.3　同一网页中的链接 …… 10
2.2.4　不同网页之间的链接 …… 11
2.3　网页的修饰 …… 12
2.3.1　字体标记 …… 12
2.3.2　页面标记 …… 14

需要全本请访问 www.ertongbook.com

2.4 多媒体页面 …… 18
2.4.1 图像标记 …… 18
2.4.2 背景音乐 …… 19
2.4.3 插入视频剪辑 …… 19
2.5 HTML 表格与表单 …… 20
2.5.1 HTML 列表 …… 20
2.5.2 HTML 表格 …… 22
2.5.3 HTML 表单 …… 23
2.6 用 FrontPage 2000 设计网页 …… 26
2.6.1 FrontPage 2000 中文版简介 …… 26
2.6.2 创建新网页和建立站点 …… 27
2.6.3 网页修饰 …… 31
2.6.4 超链接 …… 35
2.6.5 列表和表格 …… 36
2.6.6 表单 …… 37
2.6.7 动态网页 …… 39
2.6.8 框架的应用 …… 40
2.6.9 发布站点 …… 41

第 3 章 XML 语言 …… 43
3.1 从 HTML 到 XML …… 43
3.2 XML 的语法 …… 44
3.3 用 CSS 格式化 XML …… 55
3.4 用 XSL 格式化 XML …… 60

第 4 章 ASP 程序设计 …… 63
4.1 ASP 简介 …… 63
4.1.1 ASP 技术概述 …… 63
4.1.2 ASP 的工作原理 …… 65
4.1.3 ASP 运行环境 …… 67
4.1.4 ASP 程序语法简述 …… 68
4.1.5 VBScript 与 JavaScript 脚本语言 …… 74
4.2 VBScript 脚本语言 …… 75
4.2.1 VBScript 的运算符 …… 75
4.2.2 VBScript 的数据类型 …… 76
4.2.3 VBScript 常量、变量及数组 …… 77
4.2.4 VBScript 的条件语句 …… 78
4.2.5 VBScript 的循环语句 …… 79

4.2.6 VBScript 的其他语句 …… 81
4.2.7 VBScript 的过程 …… 81
4.2.8 VBScript 的基本函数 …… 83
4.3 用 ASP 实现网页互动 …… 86
4.3.1 Response 对象 …… 87
4.3.2 Request 对象 …… 89
4.3.3 使用 Cookies 集合 …… 94
4.4 网页变量的生命周期与传递 …… 97
4.4.1 提出问题 …… 97
4.4.2 Application 对象 …… 99
4.4.3 Session 对象 …… 100
4.4.4 Global.asa 文件与应用程序的初始化 …… 103
4.5 Server 对象、ODBC 与 ADO 组件 …… 106
4.5.1 Server 对象 …… 106
4.5.2 ODBC 原理与数据源的建立 …… 108
4.5.3 ADO 组件 …… 110
4.6 用 ADO 与 SQL 查询数据库 …… 119
4.6.1 建立数据库与数据源 …… 119
4.6.2 用 Select 选取所有记录 …… 120
4.6.3 用 Select 筛选记录 …… 122
4.6.4 用 Select 排序记录 …… 126
4.6.5 插入记录 …… 127
4.6.6 删除记录 …… 129
4.6.7 修改记录 …… 131
4.7 通过表单查询数据库 …… 132
4.7.1 用文本框查询数据库 …… 132
4.7.2 用复选框和下拉列表框查询数据库 …… 134
4.7.3 修改数据库中的记录 …… 137
4.8 数据库分页显示技术 …… 142
4.8.1 分页显示技术 …… 142
4.8.2 分页显示实例与分析 …… 145
4.9 数据的有效性验证 …… 149
4.9.1 在客户端验证 …… 149
4.9.2 在服务器端验证 …… 150

第 5 章 PHP 程序设计 …… 154
5.1 PHP 概述 …… 154
5.1.1 PHP 与 PHP 技术特点 …… 154

5.1.2 建立 PHP 运行环境 …… 155
5.1.3 PHP 的工作原理 …… 158
5.1.4 第一个 PHP 程序 …… 159
5.1.5 PHP 编辑器 …… 161
5.2 PHP 脚本语法 …… 162
5.2.1 数据类型 …… 162
5.2.2 PHP 变量 …… 163
5.2.3 运算符与表达式 …… 166
5.2.4 PHP 函数库 …… 166
5.2.5 自定义函数 …… 170
5.2.6 流程控制与页面重定向 …… 171
5.3 网页互动与网页变量传递 …… 175
5.3.1 用表单传递变量 …… 175
5.3.2 用 URL 传递变量 …… 178
5.3.3 用 Session 传递变量 …… 180
5.3.4 用 Cookie 传递变量 …… 183
5.4 MySQL 数据库管理 …… 185
5.4.1 MySQL 的数据类型 …… 185
5.4.2 用 phpMyAdmin 管理数据库 …… 187
5.5 Web 数据库发布技术 …… 190
5.5.1 PHP 与 Web 数据库的连接技术 …… 191
5.5.2 用 PHP 访问 MySQL 数据库 …… 192
5.6 数据查询技术 …… 196
5.6.1 模糊查询与全名查询 …… 197
5.6.2 用列表框实现多值查询 …… 198
5.6.3 介于区间的多值查询 …… 200
5.7 数据库管理技术 …… 201
5.7.1 向数据库添加新表 …… 202
5.7.2 向 buyinfo 表添加购书订单 …… 204
5.7.3 查看数据库中的数据表 …… 208
5.7.4 建立、查看与删除数据库 …… 210
5.8 用 ODBC 访问 Web 数据库 …… 215
5.8.1 用 ODBC 连接 Web 数据库的技术 …… 215
5.8.2 ODBC 数据库函数 …… 215
5.8.3 ODBC 数据库访问实例 …… 216

第 6 章 JSP 程序设计 …… 220
6.1 JSP 技术概述 …… 220

6.1.1 什么是 JSP …… 220
6.1.2 创建 JSP 运行环境 …… 220
6.1.3 第一个 JSP 程序 …… 221
6.2 JSP 脚本语言 …… 222
6.2.1 编写第一个 Java 程序 …… 222
6.2.2 基本数据类型 …… 223
6.2.3 表达式 …… 223
6.2.4 控制语句 …… 224
6.2.5 数组 …… 226
6.2.6 类 …… 227
6.2.7 异常处理 …… 228
6.2.8 Java Applet …… 229
6.3 JSP 程序的基本语法 …… 230
6.3.1 注释 …… 230
6.3.2 声明 …… 230
6.3.3 脚本 …… 230
6.3.4 表达式 …… 231
6.3.5 page 指令 …… 231
6.3.6 include 指令 …… 232
6.3.7 include 动作 …… 233
6.3.8 forward 动作 …… 234
6.3.9 plugin 动作 …… 235
6.3.10 内部对象 …… 236
6.4 处理 HTML 表单 …… 239
6.4.1 通过表单传递参数 …… 239
6.4.2 验证表单输入数据 …… 241
6.5 JDBC 接口技术 …… 243
6.5.1 从 ODBC 到 JDBC …… 244
6.5.2 JDBC 驱动程序 …… 245
6.5.3 在 JSP 中使用数据库 …… 245
6.6 查询数据库 …… 247
6.7 插入记录 …… 249
6.8 更新记录 …… 252
6.9 删除记录 …… 256
6.10 JavaBeans …… 258
6.10.1 useBean 动作 …… 259
6.10.2 setProperty 动作 …… 260
6.10.3 getProperty 动作 …… 261
6.10.4 使用 JavaBeans 实例 …… 261

第 7 章　ColdFusion 程序设计 …… 265
7.1　ColdFusion 基础 …… 265
7.1.1　什么是 ColdFusion …… 265
7.1.2　工作原理 …… 266
7.1.3　ColdFusion 组件和安装 …… 267
7.1.4　第一个 ColdFusion 程序 …… 268
7.2　CFML 语言 …… 268
7.2.1　数据类型 …… 269
7.2.2　变量 …… 269
7.2.3　函数 …… 271
7.2.4　表达式 …… 272
7.2.5　流程控制 …… 273
7.3　ColdFusion 表单 …… 278
7.3.1　创建表单 …… 278
7.3.2　传递表单变量 …… 281
7.3.3　表单数据确认 …… 282
7.4　查询数据库 …… 284
7.4.1　定义数据库查询 …… 284
7.4.2　输出查询结果 …… 286
7.4.3　输出数据的格式化 …… 289
7.4.4　通过表单查询数据库 …… 290
7.4.5　查询循环 …… 295
7.5　插入记录 …… 297
7.5.1　使用 cfinsert 标记 …… 297
7.5.2　与 SQL insert 一起使用 cfquery …… 299
7.6　更新记录 …… 301
7.6.1　与 SQL update 一起使用 cfquery …… 301
7.6.2　使用 update …… 304
7.7　删除记录 …… 304
7.8　复杂数据类型 …… 305
7.8.1　列表 …… 305
7.8.2　数组 …… 306
7.8.3　结构 …… 307
7.9　定制标记 …… 308
7.10　应用程序框架 …… 310
7.10.1　Session 变量 …… 311
7.10.2　Application 变量 …… 312

第 8 章　综合实例 …… 315
8.1　用 ASP 实现“留言簿” …… 315
8.1.1　设计思想 …… 315
8.1.2　“聪聪留言簿”的留言过程 …… 316
8.1.3　数据库的设计与数据源的创建 …… 317
8.1.4　“聪聪留言簿”ASP 程序设计及说明 …… 317
8.2　用 PHP 实现“留言簿” …… 324
8.3　用 JSP 实现“留言簿” …… 331
8.4　用 ColdFusion 实现“留言簿” …… 336
8.5　用 ASP 实现“网上论坛” …… 339
8.5.1　设计思想 …… 339
8.5.2　数据库的设计与数据源的创建 …… 340
8.5.3　“追踪学术进步论坛”ASP 程序设计及说明 …… 341

需要全本请访问 www.ertongbook.com



随着 Internet 的迅速发展,Web 技术同样发生了较大的变化。理解和掌握 Web 技术的基础知识,对于深入学习网页制作和 Web 数据库发布技术是有益的。

1.1 Web 网页

1.1.1 什么是 Web

World Wide Web(简称 Web 或 WWW,万维网或环球网)提供了一个图形化用户界面,以浏览 Internet 上的文档,这些文档以及它们之间的链接构成了一个庞大的信息网。

Web 起源于 1989 年 3 月,是由欧洲量子物理实验室 CERN(the European Laboratory for Particle Physics)开发的主从结构分布超媒体系统。1989 年为了将信息发送给在世界各地的研究人员,CERN 的 Tim B.Lee 定义了超文本系统。1992 年 1 月 Web 的第一个版本在瑞士日内瓦刚一面世,便因其独特的信息发布和获取方式而深受喜爱,访问 Web 很快成为 Internet 上最重要的应用之一。

Web 是一个全球性的通信系统,通过 Internet 这一通信基础使计算机能够相互传送基于超媒体的数据信息。根据 Tim 的定义,超文本文档存放在被称为 Web 服务器(Web Server)的计算机上,等待被访问;而运行在客户机上、用来访问 Web 的专用软件称为 Web 浏览器(Web Browser)。今天,全世界的 Web 服务器已逾百万,而运行 Web 浏览器以访问 Web 的计算机则不计其数。

Web 是信息资源的海洋,三个要素保证了人们能够方便地在这海洋中遨游:

统一的资源命名方案(如 URI)。

访问某一资源的方式,即协议(如 HTTP)。

在资源之间自由访问的方法,即超文本(如 HTML)。

Internet 发展到现在,Web 技术经历了 3 个发展阶段:

第一代,提供对静态文档(信息)的管理和访问(显示)。

第二代,与数据库系统相结合,提供对动态文档(信息)的访问和显示。

第三代,除动态文档生成和访问之外,还提供基于 Web 的联机事务处理能力。

1.1.2 静态网页

在网站建设发展的初期,人们全部使用 HTML(Hypertext Markup Language,超文

本标记语言)设计网页。这些简单的 HTML 文档在被手工编辑完成之后保存在与 Internet 相连接的计算机内,任何连入互联网的用户都可以访问其中的信息,这种形式的页面称做静态页面。由于 HTML 采用了超文本链接的方式,因此访问者可以很方便地从一个页面跳转到另一个页面,实现非线性浏览。正是由于这种全新的、更加合理、便捷的信息查询方式,促成了 Web 空前快速的发展。

随着互联网技术的不断发展以及网上信息呈几何级数的增加,人们逐渐发现手工编写包含所有信息和内容的页面对人力和物力都是一种极大的浪费,而且几乎已经变得难以实现。此外,采用静态页面方式建立起来的站点只能简单地根据用户的请求传送现有页面,而无法实现各种动态的交互功能。具体来说,静态页面在以下几个方面都存在明显的不足:

无法支持后台数据库。随着网上信息量的增加,以及企业和个人希望通过网络发布产品和信息的需求的增强,人们越来越需要一种能够通过简单的 Web 页面访问服务器端后台数据库的方式。这是静态页面远远不能实现的。

无法有效地对站点信息进行及时的更新。用户如果希望对传统静态页面的内容和信息进行更新或修改的话,只能够采用逐一更改每个页面的方式。在互联网发展初期网上信息较少的时代,这种做法还是可以接受的。但是现在即使是网友们的个人站点也包含着各种各样的丰富内容,因此,如何及时、有效地更新页面信息也成为一个亟待解决的问题。

无法实现动态显示效果。所有的静态页面都是事先编写好的,是一成不变的,因此访问同一页面的用户看到的都将只是相同的内容,静态页面无法根据不同的用户做不同的页面显示。

1.1.3 动态网页

“交互”是电子媒体的灵魂。通过在传统的静态页面中加入各种程序和逻辑控制,在网络的客户端和服务器端实现了动态和个性化的交流与互动。人们将这种交互性的网页称为动态网页。动态网页可以根据不同用户的不同需求,经过处理,把满意的结果返回给用户,这是与静态网页的最大区别。

有时候人们在网上浏览网页时,如果出现动态图像、滚动字幕、网站记数器等,就说网页是动态的,其实这是狭义地对动态网页的理解。

1.1.4 浏览器/服务器模式

随着 Web 技术的发展,用户足不出户仅用一台微机就可遍访世界各地的信息源。人们开始借助于 Web 技术,以获得一个更简单、更经济的解决方案,于是出现了浏览器/服务器模式,它具有客户机/服务器模式的所有优点而无其缺点。浏览器能从内部和外部服务器上获得信息,而服务器可以在任何位置运行在不同的操作系统上,通过浏览器为用户提供多种形式的信息。

浏览器/服务器模式的通信方式为:

(1) 用户打开计算机的浏览器;

(2) 输入一个网址,浏览器将生成一个请求并把它发送到指定的 Web 服务器;
(3) 服务器将主页发回,浏览器将其显示在屏幕上。

1.2 Web 数据库

1.2.1 什么是 Web 数据库

随着 Internet 技术与 Web 技术的蓬勃发展,人们已不满足于只在 Web 浏览器上获取静态的信息,人们需要通过它发表意见、查询数据,甚至进行网上购物,这就迫切需要实现 Web 与数据库的互连。Web 技术发展到今天,人们已经可以把数据库技术引入到 Web 系统中。数据库技术发展比较成熟,特别适用于对大量的数据进行组织管理,Web 技术具有较佳的信息发布途径,这两种技术的天然互补性决定了相互融合是其发展的必然趋势。将 Web 与数据库结合起来,不仅把 Web 与数据库的所有优点集中在一起,而且充分利用了大量已有的数据库信息资源,可以使用户在 Web 浏览器上方便地检索和浏览数据库的内容,这对许多软件开发者来说具有极大的吸引力。所以,将 Web 技术与数据库相结合,开发动态的 Web 数据库应用已成为当今 Web 技术研究的热点。

与传统方式相比,通过 WWW 访问数据库的优点在于:

借用现成的浏览器软件,无需开发数据库前端。如果能够通过 WWW 来访问数据库,我们就不需要开发客户端的程序,基于数据库的应用都可以通过浏览器来实现,界面统一,也减少了培训费用,能使广大用户很方便地访问数据库信息。

标准统一,开发过程简单。HTML 是 WWW 信息的组织方式,是一种国际标准,使用的 WWW 服务器与浏览器均遵循这个标准。使用 HTML 标准,开发者甚至只需学习 HTML 一种语言,使用者只需学习一种界面——浏览器界面。

交叉平台支持。几乎在各种操作系统上都有现成的浏览器可供使用,为一个 WWW 服务器书写的 HTML 文档,可以被所有平台的浏览器所浏览,实现了跨平台操作。

1.2.2 Web 数据库访问原理

对后台数据库的支持可以说是目前网络编程技术发展的趋势和最重要的应用之一。一般来说,网络编程技术都是通过提供对 ODBC(Open Database Connectivity,开放数据库连接标准)的支持,使用 SQL(Structured Query Language,结构化查询语言)实现对当前各种主流数据库的访问、查询和更新等操作。

一般实现 Web 数据库系统的连接和应用可采取两种方法,一种是在 Web 服务器端提供中间件来连接 Web 服务器和数据库服务器,另一种是把应用程序下载到客户端并在客户端直接访问数据库。中间件负责管理 Web 服务器和数据库服务器之间的通信并提供应用程序服务,它能够直接调用外部程序或脚本代码来访问数据库,因此可以提供与数据库相关的动态 HTML 页面,或执行用户查询,并将查询结果格式化成 HTML 页面,通过 Web 服务器返回给 Web 浏览器。最基本的中间件技术有公共网关接口 CGI 和应用程

序编程接口 API 两种。

Web 数据库技术采用三层或多层体系结构,前端采用基于瘦客户机的浏览器技术,通过 Web 服务器及中间件访问数据库,如图 1.1 所示。

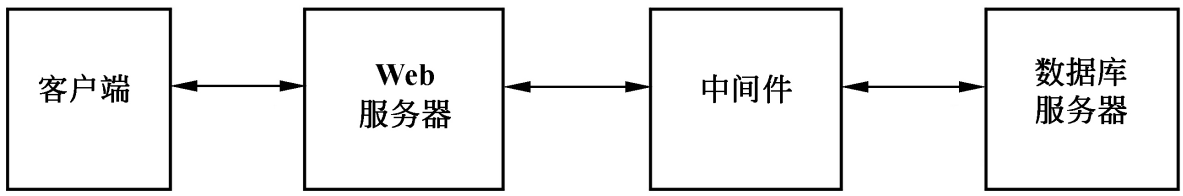


图 1.1 Web 数据库体系结构

1.2.3 Web 数据库访问技术

到目前为止,Web 数据库访问技术主要分为两大类:

公共网关接口技术(CGI);

服务器端脚本编程技术。

1. CGI

CGI 是 Web 服务器运行时外部程序的规范,按照 CGI 编写的程序可以扩展服务器的功能,完成服务器本身不能完成的工作,外部程序执行时可以生成 HTML 文档,并将文档返回 Web 服务器。CGI 应用程序能够与浏览器进行交互作用,还可以通过数据库的 API 与数据库服务器等外部数据源进行通信,如一个 CGI 程序可以从数据库服务器中获取数据,然后格式化为 HTML 文档后发送给浏览器,也可以将从浏览器获得的数据放到数据库中。几乎所有的服务器软件都支持 CGI,开发者可以使用任何一种 Web 服务器内置语言编写 CGI,其中包括流行的 C、C++、VB 和 Delphi 等。按照应用环境的不同,CGI 又可以分为标准 CGI 和间接 CGI。

2. 服务器端脚本编程技术

当前较为流行的几种有代表性的服务器端的脚本编程技术是 ASP(Active Server Pages)、PHP(Hypertext Preprocessor)、JSP(Java Server Pages)和 ColdFusion。

ASP 内含于 IIS(Internet Information Server,因特网信息服务系统)当中,提供一个服务器端的脚本(Scripting)环境,站点服务器会自动将所设计的 ASP 程序码解释为标准 HTML 格式的主页内容,在客户端的浏览器上显示出来。客户端只要使用常规可执行 HTML 码的浏览器即可浏览。

PHP 是一种跨平台的服务器端的嵌入式脚本语言。它大量地借用 C、Java 和 Perl 语言的语法,并结合 PHP 自己的特性,使 Web 开发者能够快速地生成动态页面。它支持目前绝大多数数据库。PHP 是完全免费的,可以自由下载,甚至可以不受限制地获得源代码,从而加进自己需要的特色。

JSP 是 Sun 公司推出的新一代站点开发语言,它完全解决了目前 ASP、PHP 的一个通病——脚本级执行。JSP 可以在 Serverlet 和 JavaBean 的支持下,完成功能强大的站点程序。

ColdFusion 是 Allaire 公司在 1995 年推出的产品。在目前众多的 Web 开发工具中,

简单和基于标签的语法框架使得 ColdFusion 变得简单易学。ColdFusion 拥有一个功能强大的集成开发环境(IDE),能够满足需要承受最大工作负载的商业性站点的要求,从而使 ColdFusion 的制作和开发更加快捷。

ASP、PHP、JSP 和 ColdFusion 各有所长,所以目前使用这四种技术建立网站各有自己的市场,用户可选择一种适合自己的技术来学习。

1.3 使用 SQL 语言

SQL 是数据库的标准语言。它的功能包括创建数据库对象,操纵对象,往数据库表中填充数据,在数据表中更新已存在的数据,删除数据,执行数据库查询,控制数据库访问权限和数据库总体管理。在 Web 网页中,无论何时要访问一个数据库,都要使用 SQL 语言。学好 SQL 语言,是进行 Web 数据库发布的基础。

在 SQL 语言中,最重要的概念就是查询。查询可以分为两类:选择查询和动作查询。

选择查询　该查询返回一个记录集,即按一定的条件从数据库中返回数据。

动作查询　动作查询并不返回记录集,只是对数据库中的数据进行修改,或建立和修改数据库结构,例如删除记录、建立数据库等。

1.3.1 创建表

Create Table 语句是用来在数据库中创建表,同时建立相关字段及其数据类型。虽然创建表的工作由 Create Table 一条语句就可完成,但是在 Create Table 语句真正执行前,需要花大量的时间和精力来组织表的结构。

创建表的基本语法如下:

```
Create Table table(field1 type[(size)],field2 type[(size)], ...)
```

其中,table 参数是要建立的数据表名;field1、field2 等是该数据表中相关字段的字段名;type 为相关字段的数据类型;size 则是文本字段的字符串长度,是可选参数。

例如,建立一个“学生”表,它由学生号 Stu_id、姓名 Stu_name、性别 Stu_sex、年龄 Stu_age、所在班组 Stu_class 这些字段组成。所用语句如下:

```
Create Table Student
(Stu_id CHAR(5),
Stu_name CHAR(8),
Stu_sex CHAR(2),
Stu_age INT,
Stu_class CHAR(10))
```

1.3.2 数据查询

SQL 的主要功能之一是实现数据库查询,SQL 语言提供了 Select 语句用于数据库的查询,该语句具有灵活的使用方法和丰富的功能。

在众多的 SQL 命令中,Select 语句应该算是使用最频繁的。Select 语句主要用来对数据库进行查询并返回符合用户查询标准的结果数据。Select 语句的语法格式如下:

```
Select 列表 1 [, 列表 2, ...] From 表名 [Where 条件表达式]
```

其中[ ]表示可选项。

Select 语句中位于 Select 关键词之后的列表名用来决定哪些列将作为查询结果返回。用户可以按照自己的需要选择任意列,还可以使用通配符“ * ”来设定返回记录中的所有列。

Select 语句中位于 From 关键词之后的表名用来决定将要进行查询操作的目标表格。

Select 语句中的 Where 可选从句用来规定哪些数据值或哪些行将作为查询结果返回或显示。

在 Where 条件从句中可以使用以下一些运算符来设定查询标准: = (等于)、> (大于)、< (小于)、> = (大于等于)、< = (小于等于)、< > (不等于)。

除了上面所提到的运算符外,Like 运算符在 Where 条件从句中也非常重要。Like 运算符的功能非常强大,通过使用 Like 运算符可以设定只选择与用户规定格式相同的记录。此外,我们还可以使用通配符“ % ”来代替任何字符串。

例如:

```
Select * From Student Where Stu_name Like '王%'
```

上述 SQL 语句将会查询所有以王开头的姓名。

1.3.3 插入记录

SQL 语言使用 Insert 语句向数据库表格中插入或添加新的数据行。Insert 语句的使用格式如下:

```
Insert Into 表名(字段 1,字段 2, ...,字段 n)
Values(值 1,值 2,…,值 n)
```

例如:

```
Insert Into Student  (Stu_id, Stu_name, Stu_sex, Stu_age, Stu_class)
Values( '99011', '李明', '男', '19', '98 计本')
```

简单地说,当向数据库表格中添加新记录时,在关键词 Insert Into 后面输入所要添加的表格名称,然后在括号中列出将要添加新值的列的名称。最后,在关键词 Values 后面按照前面输入的列的顺序对应地输入所有要添加的记录值。

1.3.4 删除记录

SQL 语言使用 Delete 语句删除数据库表格中的行或记录。Delete 语句的格式为:

```
Delete From 表名 Where 条件
```

需要全本请访问 www.ertongbook.com