<<项目中的.NET>>

图书基本信息

书名:<<项目中的.NET>>

13位ISBN编号:9787121176708

10位ISBN编号:712117670X

出版时间:2012-8

出版时间:电子工业出版社

作者: 李天平

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

<<项目中的.NET>>

内容概要

《项目中的.net》作者是从事.net一线开发的资深专家,常年耕耘技术博客,惠及无数.net新知。 此次将长期的思考、感悟,多年的系统开发、设计和团队管理经验,以及深入分析众多项目实战的宝 贵成果和盘托出,力求将编程思想与具体实践融为一体,提炼出适合于广大读者快速理解和彻底掌 握.net软件开发的最佳学习方法。

本书既考虑到实际开发中经常遇到的困惑和难题,也分析了解决问题的思路和方法,更总结出项目开发中不可或缺的技术点及思想。

读者可以在欣赏一个个有趣例子的过程中,不知不觉具备开发真正的商业项目的能力。

《项目中的.net》集实用性、思想性、趣味性于一体,内容共分为技术基础总结、系统架构设计思想及项目实战解析三部分。

随书所附光盘收录了大量实例代码及一些商业系统,供读者参考学习。

光盘\code\ch19

tools\动软代码生成器最新版的安装文件请到官网(http://www.maticsoft.com/)下载。

《项目中的.net》适合于.net初、中级开发人员参考学习,特别有助于.net从业者进入实战层次,也可作为相关培训机构的参考教材,还可以作为高等院校相关专业师生的参考书。

《项目中的.net》为《.net深入体验与实战精要》一书的修订版本,请已阅读过上一版的读者注意

<<项目中的.NET>>

作者简介

李天平,动软卓越(北京)科技有限公司创始人,资深互联网公司技术专家,51Aspx高级技术顾问, 具有10年以上软件及互联网产品开发经验,8年以上的项目管理和团队管理经验。

曾任红孩子集团高级开发经理,蚂蚁互动技术总监,豪杰(超级解霸)公司研发经理等职务。 从事过多年的需求分析、产品策划和架构设计工作,具有丰富的系统研发经验及项目管理和团队领导 经验。

同时也是国内著名开源软件"动软代码生成器"的作者,该软件获得国家版权局的《软件著作权》认证,下载量已经超过70万人次,著有本书前身《亮剑.NET:.NET深入体验与实战精要》一书。

<<项目中的.NET>>

书籍目录

第1篇 技术基础总结

第1章 .net,你真的知道了吗3

- 1.1 搞清自己是干什么的 4
- 1.2 .net的几个特性 7
- 1.3 万丈高楼平地起:面试者必会9
- 1.3.1 c#介绍 10
- 1.3.2 命名空间 10
- 1.3.3 c#语法格式要点 10
- 1.3.4 变量 12
- 1.3.5 类型推断 12
- 1.3.6 变量的作用域 13
- 1.3.7 常量 16
- 1.3.8 流程控制 16
- 1.3.9 字符串常见操作 21
- 1.3.10 几个常用的数学函数 26
- 1.4 .net的面向对象之门 27
- 1.4.1 继承——"子承父业"27 1.4.2 委托——"任务书"35
- 1.4.3 反射——"解剖"42
- 1.5 .net开发几把小刀 45
- 1.5.1 using之多变身 45
- 1.5.2 @符号的妙用 47
- 1.5.3 预处理指令,有你更轻松 48
- 1.6 visual studio.net 2010 实战 50
- 1.6.1 如何创建asp.net项目 51
- 1.6.2 如何创建windows项目 53
- 1.6.3 visual studio.net 2008/2010操作与使用技巧 54
- 1.6.4 常见开发调试技巧 58
- 1.6.5 错误异常处理方法 63
- 本章常见技术面试题 69
- 常见面试技巧之面试前的准备 69
- 本章小结70
- 第2章 细节决定成败 71
- 2.1 equals()和运算符==的区别 72
- 2.2 const和readonly的区别74
- 2.3 private、protected、public和internal的区别78
- 2.4 sealed、new、virtual、abstract与override 79
- 2.5 abstract class与interface 83
- 2.6 公共变量与属性的区别 85
- 2.7 参数修饰符params、out和ref的区别87
- 2.8 值类型和引用类型的区别 91
- 2.9 结构与类的区别 94
- 2.10 dispose()和close()、finalize()的区别 97
- 2.11 string和stringbuilder的区别 98
- 2.12 debug版本和release版本的区别 99

<<项目中的.NET>>

本章常见技	技术面	i试题	102
-------	-----	-----	-----

常见面试技巧之细节决定成败 102

本章小结 103

第3章 asp.net开发大杂烩 105

- 3.1 页面生命周期 106
- 3.1.1 独立页面生命周期事件顺序 106
- 3.1.2 具有master页的生命

周期事件顺序 108

- 3.1.3 asp.net生命周期详解 110
- 3.2 页面状态管理 112
- 3.2.1 cookie 113
- 3.2.2 htmlinputhidden隐藏域 115
- 3.2.3 viewstate 115
- 3.2.4 查询字符串request 116
- 3.2.5 application对象 117
- 3.2.6 session对象 117
- 3.2.7 示例项目:在线用户列表统计 120
- 3.3 服务器和客户端数据交互 123
- 3.3.1 页面数据绑定全攻略 123
- 3.3.2 bind和eval的区别 132
- 3.4 asp.net编程中的技巧 134
- 3.4.1 页面之间传值的7种方法 134
- 3.4.2 get与post方法的区别 137
- 3.4.3 asp.net服务器控件和html控件的区别 138
- 3.4.4 server.transfer和response.redirect的区别 140
- 3.4.5 刷新页面的方法汇总 141
- 3.4.6 页面事件控制 142
- 3.4.7 在url中传递中文的解决方案 144
- 3.4.8 增强用户体验的一些技巧 145
- 3.4.9 xhtml与html的区别 148
- 3.5 打造自己的页面基类pagebase 153
- 本章常见技术面试题 156
- 常见面试技巧之如何做好自我介绍 156

本章小结 157

第4章 windows窗体编程你也行 159

- 4.1 创建简单的winform项目 160
- 4.2 创建mdi窗体应用 162
- 4.3 获取应用程序路径信息 166
- 4.4 回车跳转控件焦点 167
- 4.5 窗体间传递复杂数据 168
- 4.5.1 构造传递 168
- 4.5.2 公有字段传递 169
- 4.5.3 委托与事件传递 171
- 4.6 实现个性化窗体界面 172
- 4.7 无标题窗体拖动的两种方法 175
- 4.8 让程序只启动一次——单实例运行 176
- 4.9 实现系统托盘和热键呼出 177

- 4.10 进程与多线程的区别 181
- 4.11 创建多线程应用程序 182
- 4.12 winform开发常见问题 185
- 4.12.1 如何设置运行时窗体的起始位置 186
- 4.12.2 如何使一个窗体在其他窗体的上面 186
- 4.12.3 实现窗体渐显效果 186
- 4.12.4 设置窗口背景为渐变色 187
- 4.12.5 模态窗口和非模态窗口 187
- 4.12.6 屏蔽窗口右上角的关闭 操作 188
- 4.12.7 调用执行外部的程序 188
- 本章常见技术面试题 189
- 常见面试技巧之经典问题巧回答 189
- 本章小结 190
- 第5章 数据库开发 191
- 5.1 ado.net与抽水的故事 192
- 5.1.1 ado.net的定义 192
- 5.1.2 趣味理解ado.net对象模型 194
- 5.1.3 进水笼头——建立connection 196
- 5.1.4 抽水机——command 198 5.1.5 输水管——dataadapter 201 5.1.6 输水管——datareader 202
- 5.1.7 随用随关,释放资源 204
- 5.1.8 水库管理——dataset 204
- 5.1.9 水池子——datatable 206
- 5.2 数据库操作类的封装详解 209
- 5.2.1 执行命令方法的封装 209
- 5.2.2 查询数据方法的封装 212
- 5.2.3 数据统计方法的封装 213
- 5.2.4 实现sqlparameter方式 214
- 5.2.5 实现多数据库的访问 215
- 5.3 常用的经典sql语句 216
- 5.4 事务处理 218
- 5.4.1 sql和存储过程级别的事务 219
- 5.4.2 ado.net级别的事务 221
- 5.4.3 asp.net页面级别的事务 222
- 5.4.4 企业级服务com+事务 224
- 5.4.5 system.transactions 事务处理 228
- 5.5 oracle开发常见问题 232
- 5.5.1 oracle和sql server的常用函数对比 232
- 5.5.2 oracle和sql server的语句区别 236
- 5.5.3 asp.net连接oracle
- 失败的解决方法 237
- 本章常见技术面试题 238
- 常见面试技巧之经典问题巧回答 238
- 本章小结 239
- 第6章 关于xml 241
- 6.1 xml概述 242

- 6.1.1 xml和html有什么区别 242
- 6.1.2 xml的优势 242
- 6.2 文档结构与基本语法 243
- 6.3 xml操作 245
- 6.3.1 xmldocument创建xml文档 245
- 6.3.2 xmltextwriter创建xml文档 246
- 6.3.3 xmldocument读取xml文档 248
- 6.3.4 xmltextreader读取xml文档 249
- 6.3.5 在html中显示xml中的数据 250
- 6.3.6 javascript获取xml内容 251
- 6.3.7 项目案例:通用自定义xml配置类 254
- 6.4 ado.net与xml 258
- 6.4.1 读xml文档到dataset 258
- 6.4.2 dataset转为xml文档 259
- 6.5 项目案例1:实现网站的rss应用 259
- 6.6 项目案例2:在线实现rss
- 阅读器 262
- 本章常见技术面试题 267
- 常见面试技巧之经典问题巧回答 267
- 本章小结 268
- 第7章 web service开发详解 269
- 7.1 web service基本概念 270
- 7.2 web service的应用场景 270
- 7.3 创建简单的web service项目应用 272
- 7.4 web service属性介绍 274
- 7.5 asp.net如何调用web service 278
- 7.6 javascript如何调用web service 280
- 7.6.1 通过webbehavior.htc调用web service 280
- 7.6.2 通过microsoft.xmldom调用web service 282
- 7.6.3 xmlhttp post调用web service 284
- 7.6.4 soap调用web service 285
- 7.7 winform如何调用web service 286
- 7.7.1 .net的winform调用web service 286
- 7.7.2 手动发送http请求调用web service 287
- 7.8 实现异步调用web service 289
- 7.9 如何保证web service的安全 290
- 7.9.1 通过soapheader增强web service的安全性 291
- 7.9.2 采用ssl实现加密传输 294
- 7.9.3 访问ip限制 306
- 7.10 web service开发中需要注意的问题 307
- 本章常见技术面试题 311
- 常见面试技巧之经典问题巧回答 311
- 本章小结 312
- 第8章 用户体验的杀手锏——ajax 313
- 8.1 ajax概述 314
- 8.1.1 什么是ajax 314
- 8.1.2 ajax技术的核心 315

- 8.1.3 ajax的工作原理 316
- 8.1.4 ajax的优点 316
- 8.1.5 ajax的局限性 317
- 8.1.6 ajax适用场景 318
- 8.1.7 ajax不适用的场景 319
- 8.1.8 xmlhttprequest开发实例 319
- 8.2 微软vs.net的ajax开发 323
- 8.2.1 安装asp.net 2.0 ajax extensions 323
- 8.2.2 创建asp.net ajax应用程序 324
- 8.2.3 scriptmanager控件使用技巧 325
- 8.2.4 updatepanel控件使用技巧 326
- 8.2.5 asyncpostbacktrigger实现外部控件引发局部刷新 328
- 8.2.6 ajax错误处理 331
- 8.2.7 告诉用户你正在做什么——updateprogress控件 332
- 8.2.8 asp.net ajaxcontroltoolkit简介 335
- 8.3 使用第三方组件 ajax.dll 开发 337
- 8.4 使用第三方组件ajaxpro的开发 341
- 8.5 使用第三方组件magicajax的开发 341
- 8.6 ajax开发原则 342
- 本章常见技术面试题 343
- 常见面试技巧之经典问题巧回答 343
- 本章小结 344
- 第9章系统与文件操作345
- 9.1 获取系统信息 346
- 9.1.1 用systeminformation 类获取系统信息 346
- 9.1.2 用environment 类获取系统信息 347
- 9.1.3 通过wmi获取系统信息 348
- 9.1.4 用registrykey 类读取系统信息 351
- 9.1.5 用api函数获取系统信息 353
- 9.1.6 获取系统服务信息 355
- 9.2 文件操作 357
- 9.2.1 通过streamwriter类实现写文件 358
- 9.2.2 通过file类创建文件 358
- 9.2.3 通过filestream类创建文件 359
- 9.2.4 通过fileinfo类创建文件 359
- 9.2.5 追加文本 360
- 9.2.6 读取文本文件 360
- 9.2.7 读写二进制文件 362
- 9.2.8 文件复制、移动和删除 363
- 9.3 文件夹目录操作 364
- 9.4 读写ini文件 365
- 9.5 读写注册表 368
- 本章常见技术面试题 370
- 常见面试技巧之经典问题巧回答 370
- 本章小结 371
- 第10章 网络应用开发 373
- 10.1 socket基本编程 374

- 10.1.1 socket基本知识 374
- 10.1.2 socket服务端开发步骤 376
- 10.1.3 socket客户端开发步骤 378
- 10.2 异步socket通信——实现msn机器人 380
- 10.2.1 机器人服务端 380
- 10.2.2 客户端实现步骤 384
- 10.3 基于tcp协议的客户端和服务端 387
- 10.3.1 tcplistener 实现网络服务端 387
- 10.3.2 tcpclient实现网络客户端 388
- 本章常见技术面试题 390
- 常见面试技巧之经典问题巧回答390
- 本章小结 391
- 第11章 windows service开发 393
- 11.1 什么是windows服务 394
- 11.2 创建windows服务 394
- 11.3 windows服务开发常见问题 398
- 11.4 安装/卸载windows服务 399
- 11.4.1 安装windows服务 399
- 11.4.2 卸载windows服务 400
- 11.4.3 windows服务应用程序体系结构 400
- 11.5 调试windows服务 403
- 11.5.1 日志调试法 404
- 11.5.2 附加进程断点调试法 404
- 本章常见技术面试题 405
- 职场智慧之初入江湖 405
- 本章小结 407
- 第12章 新技术初探 409
- 12.1 wpf 411
- 12.1.1 wpf简介 411
- 12.1.2 wpf何以令人佩服 411
- 12.1.3 xaml概述 412
- 12.1.4 wpf开发环境配置 413
- 12.1.5 项目示例:开发一个简单的wpf应用程序 414
- 12.2 wcf 416
- 12.2.1 soa是什么 417
- 12.2.2 wcf是什么 417
- 12.2.3 wcf的优势是什么 418
- 12.2.4 wcf开发环境 421
- 12.2.5 项目示例: 订票服务wcf开发步骤 421
- 12.3 ling 430
- 12.3.1 ling的架构 430
- 12.3.2 传统的查询 431
- 12.3.3 ling查询实例 433
- 12.3.4 ling查询语法 438
- 本章常见技术面试题 439
- 职场智慧之学习方法 439
- 本章小结 440

- 第13章 面向对象思想 443
- 13.1 为什么要面向对象 444
- 13.2 什么是面向对象 446
- 13.3 面向对象的特点 448
- 13.3.1 封装 448
- 13.3.2 继承 449
- 13.3.3 多态 450
- 13.4 面向对象设计和开发实例 456
- 13.4.1 用传统的过程化设计实现播放器功能 456
- 13.4.2 基于面向对象设计实现播放器功能 457
- 13.4.3 面向对象封装、继承、多态的应用 458
- 13.4.4 设计模式、条件外置及反射技术的应用 461
- 13.5 面向对象分析(ooa)的方法 464
- 13.6 面向对象设计的原则 468
- 13.6.1 优先使用(对象)组合,而非(类)继承 468
- 13.6.2 针对接口编程,而非(接口的)实现 470
- 13.6.3 开放 封闭法则 (ocp) 472
- 13.6.4 liskov替换法则(lsp) 474
- 13.6.5 单一职责原则(srp) 475
- 13.6.6 依赖倒置原则(dip) 476
- 13.6.7 接口分离原则(isp) 478
- 13.6.8 面向对象设计时需要
- 注意的问题 479
- 本章常见技术面试题 480
- 本章小结 481
- 第14章 三层架构项目开发 483
- 14.1 什么是三层架构 484
- 14.1.1 常用的三层架构设计 484
- 14.1.2 趣味理解:三层架构与养猪 486
- 14.2 为什么要用三层架构 487
- 14.3 三层架构项目开发示例 490
- 14.3.1 数据库设计 490
- 14.3.2 创建整体解决方案 491
- 14.3.3 业务实体model的实现 492
- 14.3.4 数据访问层的实现 494
- 14.3.5 业务逻辑层的实现 498
- 14.3.6 表示层的实现 499
- 14.4 实现基于工厂模式的三层架构 501
- 14.4.1 扩展新增数据访问层 502
- 14.4.2 idal抽象接口的实现 503
- 14.4.3 创建dal对象的封装 507
- 14.4.4 实现抽象工厂模式 508
- 14.4.5 工厂模式的三层架构图 511
- 本章常见技术面试题 512
- 职场智慧之程序员的职业规划 512
- 本章小结 513

笙15音	大型网站	钓性能优	计分片安全	515
わい早	ᄉᆂᄢᄱᆈ	1ソ 1工 日匕 ル	ᆡပᅴᆺᆂ	. טוט

- 15.1 高效c#编码优化 516
- 15.2 页面 (html) 优化的方法 523
- 15.3 asp.net开发性能优化 529
- 15.3.1 尽量使用静态html页面 530
- 15.3.2 避免不必要的回送操作 530
- 15.3.3 尽量在客户端进行用户输入验证 531
- 15.3.4 关闭不必要的session状态 531
- 15.3.5 优先使用html控件,而非服务器控件531
- 15.3.6 不必要时关闭viewstate 532
- 15.3.7 禁用调试模式 532
- 15.4 系统缓存管理 533
- 15.4.1 缓存概述 533
- 15.4.2 传统缓存方式 534
- 15.4.3 页面输出缓存 534
- 15.4.4 页面输出缓存api 537
- 15.4.5 页面局部缓存 539
- 15.4.6 应用程序数据缓存 543
- 15.4.7 文件缓存依赖 545
- 15.4.8 数据库缓存依赖 546
- 15.4.9 memcached——分布式缓存系统 550 15.4.10 cacheman——.net架构下的分布式缓存项目 551
- 15.5 数据库访问性能优化 552
- 15.5.1 选择合适的.net数据供应器 552
- 15.5.2 及时关闭数据库连接 552
- 15.5.3 跟踪监视sql server当前链接池状态 553
- 15.5.4 善用数据库的存储过程 556
- 15.5.5 sqldatareader和dataset的选择 556
- 15.5.6 executenonquery和executescalar的选择 557
- 15.5.7 数据的绑定databinder 557
- 15.5.8 使用datareader的注意事项 557
- 15.5.9 command对象的使用 558
- 15.5.10 反复执行sql语句用prepare() 558
- 15.5.11 分页的数据访问 559
- 15.5.12 sql命令的优化 559
- 15.5.13 tempdb的使用技巧 561
- 15.5.14 使用视图代替跨库操作 561
- 15.5.15 尽量避免大事务操作和游标的使用 561
- 15.5.16 为表建立适当的索引 562
- 15.6 网站安全防护 562
- 15.6.1 什么是sql注入式攻击 562
- 15.6.2 如何防范sql注入式攻击 563
- 15.6.3 实现页面验证码 564
- 15.6.4 实现文件防盗链 568
- 15.6.5 采用https进行访问 572
- 15.7 iis优化 573
- 15.7.1 如何监测iis服务器并发数 573

- 15.7.2 采用gzip压缩页面优化 574
- 15.7.3 网站应用程序池配置 576
- 15.8 网站架构优化策略 581
- 本章常见技术面试题 584
- 职场智慧之独善其身 584
- 本章小结 587
- 第16章 系统设计的原则和技巧 589
- 16.1 系统设计的原则 590
- 16.1.1 最适合的才是最好的 590
- 16.1.2 以不变应万变 591
- 16.1.3 可扩展性 591
- 16.1.4 可复用性 592
- 16.2 系统设计的常用方法 592
- 16.3 敏捷软件开发12条原则 596
- 16.4 系统架构师成长之路 599
- 本章常见技术面试题 603
- 职场智慧之职场政治603
- 本章小结604
- 第3篇 项目实战解析
- 第17章 权限角色管理项目解析 607
- 17.1 权限角色管理概述 608
- 17.2 项目概述 611
- 17.3 数据库设计 612
- 17.3.1 数据库实体关系模型 612
- 17.3.2 表结构设计 613
- 17.4 数据访问层 615
- 17.4.1 类设计 615
- 17.4.2 代码实现 616
- 17.5 业务逻辑层 617
- 17.5.1 类设计 617
- 17.5.2 扩展.net framework用户处理机制 620
- 17.6 接口调用与web管理实现 625
- 17.6.1 web.config配置 626
- 17.6.2 用户身份和权限验证 626
- 17.6.3 用户验证接口方式1:权限验证用户控件 627
- 17.6.4 用户验证接口方式2:页面基类 629
- 17.6.5 用户和角色权限的后台管理 632
- 职场智慧之如何提升自己在公司的价值 639
- 本章小结 640
- 第18章 单点登录系统的设计与实现 641
- 18.1 项目概述 642
- 18.1.1 名词定义 642
- 18.1.2 项目需求描述 642
- 18.2 业务流程设计 643
- 18.2.1 用户认证流程 643
- 18.2.2 安全验证流程 644
- 18.3 功能与接口设计 645

- 18.3.1 接口交互设计 645
- 18.3.2 应用系统接口 646
- 18.3.3 认证服务器功能和接口 646
- 18.4 数据库设计 646
- 18.4.1 数据库实体关系模型 647
- 18.4.2 表结构设计 647
- 18.5 实现认证服务器 648
- 18.5.1 sso解决方案 649
- 18.5.2 系统管理后台 650
- 18.5.3 用户通行证管理中心 651
- 18.5.4 认证服务器接口开发 658
- 18.6 应用系统接入接口开发 661
- 18.6.1 用户身份认证 661
- 18.6.2 接收状态同步 662
- 18.6.3 用户注销 664
- 18.6.4 更新认证服务器状态 664
- 18.7 接口封装及调用 665
- 职场智慧之晋升之道667
- 本章小结 670
- 第19章 常用.net开发工具介绍 671
- 19.1 源码版本管理: visual sourcesafe 2005 672
- 19.1.1 安装和配置vss 2005服务端 672
- 19.1.2 安装和配置vss客户端 676
- 19.1.3 将项目加入vss版本控制 677
- 19.1.4 客户端连接vss 服务器获取源代码 679
- 19.1.5 设置vss支持通过
- internet访问 681
- 19.1.6 版本控制的几个概念 685
- 19.1.7 vss项目权限管理 687
- 19.1.8 vss数据备份 689
- 19.1.9 vss 2005的使用规范 691
- 19.2 单元测试工具: nunit 692
- 19.2.1 nunit简介 692
- 19.2.2 手把手教你在.net中应用nunit 692
- 19.3 日志工具——log4net 698
- 19.3.1 log4net简介 698
- 19.3.2 使用步骤 701
- 19.3.3 单独配置文件的使用 702
- 19.4 代码规范检查工具: microsoft source analysis for c# 705
- 19.5 生成文档注释工具:ghostdoc 707
- 19.6 反编译工具: reflector for .net 714
- 19.7 动软代码生成器 715
- 19.8 帮助文档生成工具: sandcastle 718
- 19.8.1 sandcastle介绍 719
- 19.8.2 生成方式 720
- 19.8.3 具体生成步骤 721
- 本章小结 729

<<项目中的.NET>>

附录a 软件编码规范 731

- a.1 概述 732
- a.1.1 规范基本原则 732
- a.1.2 术语定义 732
- a.1.3 文件命名组织 732
- a.2 代码外观 733
- a.2.1 列宽 733
- a.2.2 换行 733
- a.2.3 缩进 733
- a.2.4 空行 733
- a.2.5 空格 734
- a.2.6 括号——() 734
- a.2.7 花括号——{} 735
- a.3 程序注释 735
- a.3.1 注释概述 735
- a.3.2 文档型注释 736
- a.3.3 类注释 737
- a.3.4 单行注释 737
- a.3.5 注释标签 737
- a.4 声明 739
- a.4.1 每行声明数 739
- a.4.2 初始化 739
- a.4.3 位置 740
- a.4.4 类和接口的声明 740
- a.4.5 字段的声明 740
- a.5 命名规范 741
- a.5.1 命名概述 741
- a.5.2 大小写规则 742
- a.5.3 缩写 742
- a.5.4 命名空间 743
- a.5.5 类 743
- a.5.6 接口 744
- a.5.7 属性 (attribute) 744
- a.5.8 枚举 (enum) 744
- a.5.9 参数 745
- a.5.10 方法 745
- a.5.11 属性 (property) 745
- a.5.12 事件 747
- a.5.13 常量 (const) 748
- a.5.14 字段 748
- a.5.15 静态字段 749
- a.5.16 集合 749
- a.5.17 措词 749
- a.6 语句 750
- a.6.1 每行一条语句 750
- a.6.2 复合语句 750
- a.6.3 return语句 751

- a.6.4 if、if-else、if else-if语句 751
- a.6.5 for、foreach 语句 752
- a.6.6 while语句 752
- a.6.7 do-while语句 752
- a.6.8 switch-case语句 752
- a.6.9 try-catch语句 753
- a.6.10 using块语句 753
- a.6.11 goto语句 754
- a.7 其他 754
- a.7.1 表达式 754
- a.7.2 类型转换 754
- a.8 匈牙利命名法 754
- a.9 控件命名规则 755
- a.9.1 一般命名方法 755
- a.9.2 主要控件名简写对照表 755

<<项目中的.NET>>

章节摘录

版权页: 插图: 4.XML 可扩展的标记语言(Extensible Markup Language)具有一种开放的、可扩展的、可自描述的语言结构,它已经成为网上数据和文档传输的标准。

它是用来描述数据结构的一种语言,正如它的名字一样。

它使对某些结构化数据的定义更加容易,并且可以通过它和其他应用程序交换数据,具体内容可以参考第6章。

8.1.3 AJAX的工作原理 AJAX的作用相当于在用户和服务器之间加了一个中间层,使用户操作与服务器响应异步化。

这样把以前的一些服务器负担的工作转嫁到客户端,利用客户端闲置的处理能力来进行处理,从而减轻服务器和带宽的负担,达到节约带宽和提高用户体验的目的。

图8.1演示了AJAX的这种工作原理。

通过在用户和服务器之间引入一个AJAX引擎,可以消除Web的开始—停止—开始—停止这样一个交互过程。

通过JavaScript发送HttpRequest异步请求,返回数据后,由JavaScript来更新客户端页面,而在这期间 ,JavaScript可以通过动画效果或动态提示告诉用户当前的进度。

同时也避免了整个页面刷新下载所带来的资源浪费。

- 8.1.4 AJAX的优点 AJAX 的优点如下。
- (1)减轻服务器的负担,提升站点的性能。

因为AJAX的根本理念是"按需取数据",所以最大可能地减少了冗余请求和响应对服务器造成的负担(以前是服务器每次生成HTML页面并返回给客户端(浏览器))。

在大多数网站中,很多页面至少90%都是一样的,比如结构、格式、页头、页尾、广告等,不同的只是一小部分的内容,但每次服务器都会生成所有的页面,再返回给客户端,这无形中是一种浪费,不管是对用户的时间、带宽、CPU耗用,还是对ISP的高价租用的带宽和空间来说。

如果按一页来算,那么几千字节或是几十千字节可能并不起眼,但对像新浪网这样每天要生成几百万个页面的大ISP来说,其损失是巨大的。

而AJAX可以作为客户端和服务器的中间层来处理客户端的请求,并根据需要向服务器端发送请求,用什么就取什么,用多少就取多少,这不会有数据的冗余和浪费,减少了数据下载总量,节省了大量资源,减轻了服务器负担,提升了站点的性能。

(2) 无刷新更新页面,减少用户实际和心理等待时间。

<<项目中的.NET>>

编辑推荐

《项目中的.NET》适合于.NET初、中级开发人员参考学习,特别有助于.NET从业者进入实战层次,也可作为相关培训机构的参考教材,还可以作为高等院校相关专业师生的参考书。

<<项目中的.NET>>

名人推荐

从C#基础语法到WinForm开发,再到大型网站项目优化,本书在系统讲解.NET要点之余,奉上宝贵的实战经验,对有志投身于软件开发行业者的能力提升大有裨益,值得一读。

- ——搜狐首席架构师 秦东亮 作者多年经验所得,能让读者更快捷而容易地通晓原理、解决实际问题 、开发优秀应用程序。
- 本书需要结合实际开发细细品味,方能豁然开朗,领悟书中揭示的.NET技术精髓。
- ——中软公司经理 吴宝辉 在网络发达的今天,像这本书这般值得仔细研读的技术书籍可谓寥寥。
- "项目中的.NET"蕴藏多年实战磨砺而成的真知灼见,值得盘踞案头、长驻行囊。
- ——51 Aspx创始人、微软MVP 刘海峰 一本我会推荐的书,一本可助力.NET程序员飞跃的书。 作者将主题讲解得极为明白易懂地,读者既可以用来系统学习,也可作为日常工具书。
- ——中讯集团微软MVP、微软讲师 穆彦鹏 本书侧重实用,对真实的商业案例透彻剖析,更有大量精心筛选的面试题,对大学生就业及程序员求职尤为有益。
- ——华东师范大学软件学院教师博士 高岩

<<项目中的.NET>>

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