

图书基本信息

书名：<<C#网络编程高级篇之网页游戏辅助程序设计>>

13位ISBN编号：9787121177934

10位ISBN编号：7121177935

出版时间：2012-8

出版时间：电子工业出版社

作者：张慧斌

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>

内容概要

本书通过编写C#网络编程语言中具有代表性的实例，向读者深入细致地讲解了如何利用C#语言进行网页游戏辅助程序设计。

本书通过大量的代码引导读者一步步学习和掌握C#的网络应用编程的方法和网页游戏辅助程序的设计技术。

本书涉及的领域包括多线程编程技术、Socket套接字编程、TCP协议编程、HTTP协议编程、远程控制技术、木马技术、模拟键盘和鼠标技术、网页游戏辅助程序设计技术等。

书籍目录

第1章 多线程编程技术1

- 1.1 进程和线程的概念1
- 1.2 C#中进程和线程的管理6
 - 1.2.1 管理进程的Process类6
 - 1.2.2 C#多线程8
 - 1.2.3 线程池12
- 1.3 C#委托机制19
 - 1.3.1 委托的定义19
 - 1.3.2 委托的使用21
- 1.4 在线程中调用窗口控件24
- 1.5 回调机制28
 - 1.5.1 回调机制的概念28
 - 1.5.2 回调方法实现的一般过程28
 - 1.5.3 发起和完成异步调用的方案30
 - 1.5.4 多线程和方法回调的综合例子33

第2章 套接字编程35

- 2.1 套接字类简介35
 - 2.1.1 什么是套接字35
 - 2.1.2 Socket类及其常用属性36
 - 2.1.3 Socket类的常用方法38
- 2.2 IP协议和端口40
 - 2.2.1 IP协议和端口的概念40
 - 2.2.2 IPAddress类与Dns类40
 - 2.2.3 IPHostEntry类41
 - 2.2.4 IPEndPoint类41
- 2.3 TCP协议工作原理43
- 2.4 同步TCP应用编程44
 - 2.4.1 TCP应用程序编程步骤44
 - 2.4.2 TcpClient类与TcpListener类45
 - 2.4.3 同步套接字建立服务器端程序47
 - 2.4.4 同步套接字TCP客户端程序51
- 2.5 异步TCP应用编程54
 - 2.5.1 异步套接字编程的方法55
 - 2.5.2 异步套接字的服务器端程序57
 - 2.5.3 异步套接字的客户端程序62

第3章 网络编程实例68

- 3.1 网络抓包监控软件的实现68
- 3.2 网络对战游戏设计79
- 3.3 即时通信软件的设计100

第4章 HTTP编程技术139

- 4.1 HTTP协议详解139
 - 4.1.1 HTTP协议的URL141
 - 4.1.2 HTTP协议的请求格式141
 - 4.1.3 HTTP协议的响应格式142
 - 4.1.4 HTTP协议的其他报头格式143

- 4.1.5 一个HTTP协议的完整通信过程144
- 4.2 WebBrowser使用实例145
- 4.3 HTTP协议编程实例148
 - 4.3.1 C#实现HTTP协议多线程文件传输148
 - 4.3.2 简易网络蜘蛛程序的实现153
- 第5章 远程控制编程技术176
 - 5.1 远程控制的原理和概念176
 - 5.2 一个简单的远程控制软件的实现177
 - 5.3 木马的原理和特性193
 - 5.3.1 木马的原理193
 - 5.3.2 木马的特性195
 - 5.4 一个简单木马程序的实现196
 - 5.4.1 木马的服务端程序196
 - 5.4.2 木马的控制端程序200
- 第6章 简单游戏辅助程序设计205
 - 6.1 模拟鼠标键盘的辅助程序设计技术205
 - 6.1.1 键盘模拟技术206
 - 6.1.2 鼠标模拟技术215
 - 6.2 基于封包技术的辅助程序设计218
 - 6.2.1 获取网络游戏的协议数据包的消息格式218
 - 6.2.2 QQ农场辅助程序的设计思路219
 - 6.2.3 QQ农场辅助程序的实现219
- 第7章 网页游戏热血三国之辅助程序设计267
 - 7.1 网页游戏辅助程序设计的必备基础知识268
 - 7.1.1 如何解析AMF协议数据包消息268
 - 7.1.2 使用Winpcap从网卡直接获取数据包269
 - 7.1.3 使用SWFDecompiler的SWF文件反编译工具获得网页游戏源码270
 - 7.1.4 网页游戏的加/解密方法270
 - 7.2 网页游戏热血三国辅助程序的实现271
- 参考文献342

章节摘录

版权页：插图：与同步工作方式和异步工作方式相对应，利用Socket类进行编程时系统也提供相应的方法，采用相应的方法进行编程分别称为同步套接字编程和异步套接字编程。

但是使用套接字编程比较复杂，涉及到很多底层的细节。

为了简化套接字编程，.NET框架又专门提供了两个类：TcpClient类与TcpListener类。

由于这两个类与套接字一样也分别有各自的同步和异步工作方式及其对应的方法，而我们在编程时，三个类都有可能使用，因此为简化起见，无论使用的是哪个类，我们统统从工作方式上将其称为同步TCP和异步TCP，所以其编程方式也有两种，一种是同步TCP编程，另一种是异步TCP编程。

注意这里的同步TCP和异步TCP仅仅指工作方式，它和线程间的同步不是一个概念。

线程间的同步是指不同线程或不同线程使用的某些资源具有先后关联的关系，它决定着逻辑执行的顺序。

比如有A和B两个资源，实际应用中要求只有对资源A处理后才能处理资源B，就说A和B存在同步关系。

如果执行顺序不正确，变为先处理资源B再处理资源A，得到的结果就是错误的。

所以，线程间的同步主要关注的是一种逻辑关系。

而同步TCP和异步TCP则仅仅指TCP协议编程中采用哪种工作方式而言，即是从执行到发送、接收和监听语句时，程序是否继续往下执行这个角度说的。

从逻辑关系上看，无论是同步TCP应用编程还是异步TCP应用编程，在实际应用中可能要求不同线程间的同步，也可能不要求同步。

要进行C#基于TCP协议的网络通信，首先必须建立同远程主机的连接，连接地址通常包括两部分——主机名和端口，如“www.sohu.com:80”中，www.sohu.com就是主机名，80指主机的80端口。

当然，主机名也可以用IP地址代替。

当连接建立之后，就可以使用这个连接来发送和接收数据包，TCP协议的作用就是保证这些数据包能到达终点，并且能按照正确的顺序组装起来。

在.NET framework的类库（Class Library）中，提供了两个用于TCP网络通信的类，分别是TcpClient和TcpListener。

由其英文意义显而易见，TcpClient类是基于TCP协议的客户端类，而TcpListener是服务器端类，监听（Listen）客户端传来的连接请求。

TcpClient类通过TCP协议与服务器进行通信，并获取信息，它的内部封装了一个Socket类的实例，这个Socket对象被用来使用TCP协议向服务器请求和获取数据。

因为与远程主机的交互是以数据流的形式出现的，所以传输的数据可以使用。

2.4 同步TCP应用编程 2.4.1 TCP应用程序编程步骤 TCP应用程序是基于C/S模式的，双方通信的最基本前提就是客户端要先和服务器端进行TCP连接，然后才可以在此基础上相互收发数据。

在服务器端，程序员需要编写程序来不断地监听客户端是否有连接请求，并通过套接字区分是哪个客户；而客户端与服务器连接则比较简单，只需要指定连接的是哪个服务器即可。

一旦双方建立了连接，并创建了对应的套接字，就可以相互收发数据。

在程序中，发送和接收数据的方法都是一样的，区别仅是方向的不同。

在同步TCP应用编程中，发送、接收和监听语句均采用阻塞方式进行工作。

使用同步TCP编写服务器端程序的一般步骤为：（1）创建一个包含采用网络类型、数据传输类型和协议类型的本地套接字对象，并将其与服务器的IP地址和端口号绑定。

这个过程可以通过Socket类或者TcpListener类完成。

（2）在指定的端口进行监听，以便接受客户端连接请求。

编辑推荐

《C#网络编程高级篇之网页游戏辅助程序设计》可作为高等院校计算机及相关专业学生的C#语言课程实践教材，也适合有一定编程基础向更高级编程阶段进阶的程序开发人员阅读，更是设计游戏辅助程序的开发人员不可多得的一本参考教材。

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>