

C# 网络编程之Tcp实现客户端和服务端聊天

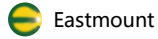
原创 Eastmount 最后发布于2013-07-20 13:26:11 阅读数 15573 ☆ 收藏

展开



Python+TensorFlow人工智能

该专栏为人工智能入门专栏，采用Python3和TensorFlow实现人工智能相关算法。前期介绍安装流程、基础语法...



¥9.90

去订阅

最近使用Socket网络套接字编程中，在同步与异步通讯中客户端与服务器总是无法响应，但在学习Tcp协议编程中完成了通讯聊天功能，下面简单讲讲我最近学到的及Tcp聊天的源代码及详细注释。

Tcp协议是一个传输层的协议，在Tcp协议编程中它通常使用的是3个类，其命名空间为System.Net.Sockets：

1.TcpListener:基于TCP协议服务端开发,监听IP地址和端口号是否连接。

该类常用的方法有Start()开始监听、AcceptSocket()返回套接字接受连接请求、AcceptTcpClient()返回客户对象接受连接请求、Stop()停止监听

2.TcpClient:基于TCP协议客户端编程,提供客户端连接，通过网络连接发送接受数据。

该类常用的方法有Connect()与服务器主机连接、GetStream()用来获得答应的数据流、Close()关闭连接

3.NetWorkStream:用于获取和操作网络流，该程序中还是用写入流和读取流对象实现写入和读取数据的操作。

该类常用的方法有Read()从网络流中读取数据、Write()从网络流中写数据。

下面是这个例子的源代码：（该代码是根据自己所学及《C#范例开发大全》中一个例子改变而来通过两个控制台程序实现通话）

1.服务端代码(TCPServer)

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

// 添加新的命名空间
using System.Net;
using System.Net.Sockets;
using System.IO;           // 流StreamReader

namespace TCPServer
{
    class Program
    {
        static void Main(string[] args)
        {
            //Parse将字符串转换为IP地址类型
            IPAddress myIP = IPAddress.Parse("127.0.0.1");
            // 构造一个TcpListener(IP地址, 端口) 对象,TCP服务端
            TcpListener myServer = new TcpListener(myIP, 6688);

            // 开始监听
            myServer.Start();
            Console.WriteLine("等候一个连接...");

            // 构造TCP客户端: 接受连接请求
            TcpClient client = myServer.AcceptTcpClient();
            Console.WriteLine("客户端已经连接...");
        }
    }
}
```

```

// 构造NetworkStream类, 该类用于获取和操作网络流
// 读数据流对象
StreamReader sr = new StreamReader(stream);
// 写数据流对象
StreamWriter sw = new StreamWriter(stream);

while (true)
{
    Console.WriteLine("客户端:" + sr.ReadLine());
    string msg = Console.ReadLine();
    sw.WriteLine(msg);
    sw.Flush();           // 刷新流
}
client.Close();          // 关闭客户端
}
}
}

```

2.客户端代码(TCPClient)

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

// 添加新的命名空间
using System.Net;
using System.Net.Sockets;
using System.IO;           // 流StreamReader

namespace TCPClient
{
    class Program
    {
        static void Main(string[] args)
        {
            // Parse将字符串转换为IP地址类型
            IPAddress myIP = IPAddress.Parse("127.0.0.1");
            // 构造一个TcpClient类对象, TCP客户端
            TcpClient client = new TcpClient();
            // 与TCP服务器连接
            client.Connect(myIP, 6688);
            Console.WriteLine("服务器已经连接...请输入对话内容...");

            // 创建网络流, 获取数据流
            NetworkStream stream = client.GetStream();
            // 读数据流对象
            StreamReader sr = new StreamReader(stream);
            // 写数据流对象
            StreamWriter sw = new StreamWriter(stream);

            while (true)
            {
                string msg = Console.ReadLine();
                sw.WriteLine(msg);
                sw.Flush();           // 刷新流
                Console.WriteLine("服务器:" + sr.ReadLine());
            }
        }
    }
}

```

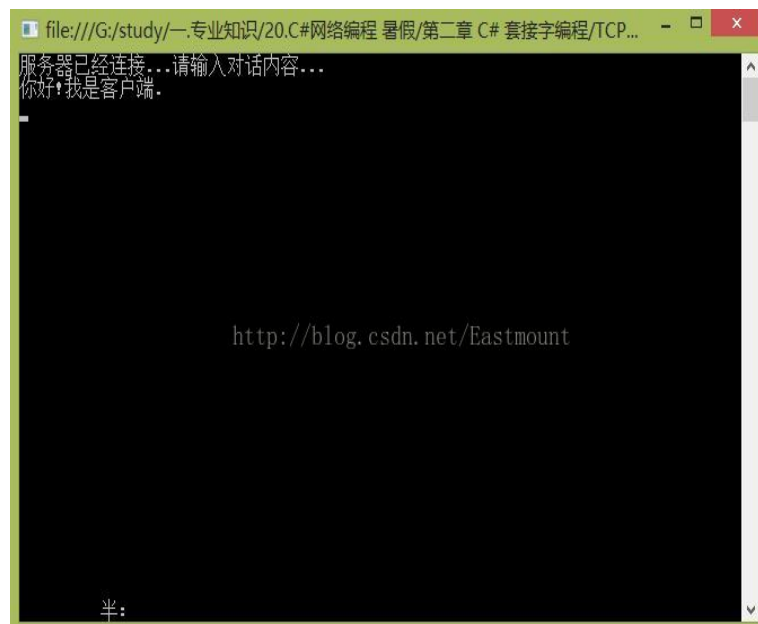
```

        client.Close();
        Console.Read();
    }
}
}

```

3.程序演示

先运行服务器(TCPServer)代码, 它会显示 "等候一个连接..." .再运行客户端(TCPClient)代码, 运行后此时服务端显示"客户端已连接...",客户端显示"服务器已连接...请输入对话内容".然后依次在客户端和服务器中个输入聊天内容,在另一方会显示相应传输过来的内容, 实现TCP聊天通话。下面是在客户端输入"你好!我是客户端."的反应。



更多的通话内容如下所示:

```
file:///G:/study/一.专业知识/20.C#网络编程 暑假/第二章 C# 套接字编程/TCP...
等候一个连接...
客户端已经连接...
客户端:你好!我是客户端.
你好!我是服务器.
客户端:Client
Server
客户端:123456
abcde
客户端:非常有趣的例子
赞同

http://blog.csdn.net/Eastmount

半:
```

```
file:///G:/study/一.专业知识/20.C#网络编程 暑假/第二章 C# 套接字编程/TCP...
服务器已经连接...请输入对话内容...
你好!我是客户端.
服务器:你好!我是服务器.
Client
服务器:Server
123456
服务器:abcde
非常有趣的例子
服务器:赞同

http://blog.csdn.net/Eastmount

半:
```

总结:这是自己认为写得比较好的一篇文章,它比较实用,更重要的是我编写上面的代码能运行。我们只需要稍微的界面设计就能实现一个有界面的网络聊天室.它涉及到的知识主要是C#网络编程中的TCP编程知识,最近写Socket套接字方面的通讯总是失败,也看了很多博客下载了很多程序,但都没一个比较满意的。希望上面的文章能帮组到大家,有错误不足之处见谅!

最后感谢刘丽霞的《C#范例开发大全》和周存杰的《C#网络编程实例教程》两本书,上面的知识点和代码都是借鉴了他们前人的知识,希望大家也看看这两本比较好的C#书。

(By:Eastmount 2013-7-20 <http://blog.csdn.net/eastmount/>)

点赞 2 ☆ 收藏 分享 ...



Eastmount 博客专家

发布了446 篇原创文章 · 获赞 6052 · 访问量 489万+

他的留言板

关注

