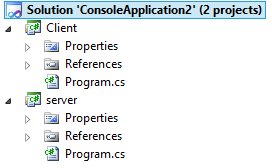
Multi-client per one server - socket programming in .net(C#) Works for me

**How to develop a socket server program which can be able to be connected by as much client as it wish!**"

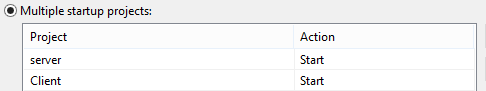
The solution contains two project:

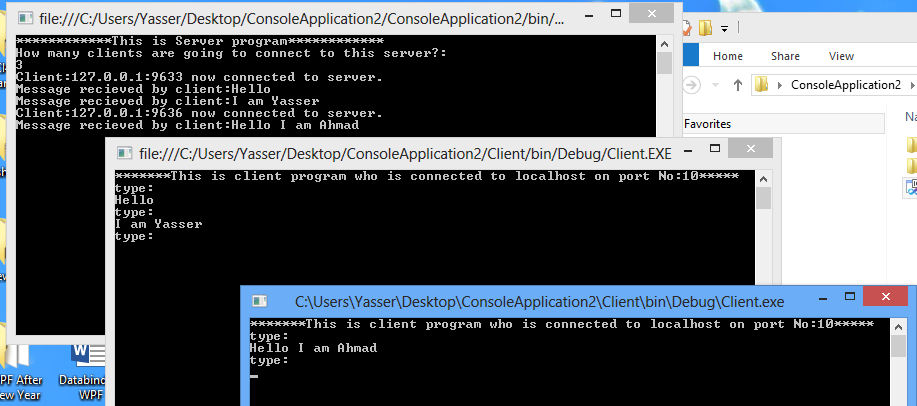
1. Server and
2. Client

Both of them are console application (with .NET framework 2.0 developed with Visual Studio 2010). First I will demonstrate the Server codes and then the Client code since the later is much more simple



Right click Solution, Properties:





using System;

using System.Net.Sockets;

using System.Threading;

public class AsynchIOServer

{

static TcpListener tcpListener = new TcpListener(10);

static void Listeners()

{

Socket socketForClient = tcpListener.AcceptSocket();

if (socketForClient.Connected)

{

Console.WriteLine("Client:"+socketForClient.RemoteEndPoint+" now connected to server.");

NetworkStream networkStream = new NetworkStream(socketForClient);

System.IO.StreamWriter streamWriter =

new System.IO.StreamWriter(networkStream);

System.IO.StreamReader streamReader =

new System.IO.StreamReader(networkStream);

while (true)

{

string theString = streamReader.ReadLine();

Console.WriteLine("Message recieved by client:" + theString);

if (theString == "exit")

break;

}

streamReader.Close();

networkStream.Close();

streamWriter.Close();

}

socketForClient.Close();

Console.WriteLine("Press any key to exit from server program");

Console.ReadKey();

}

public static void Main()

{

tcpListener.Start();

Console.WriteLine("\*\*\*\*\*\*\*\*\*\*\*\*This is Server program\*\*\*\*\*\*\*\*\*\*\*\*");

Console.WriteLine("How many clients are going to connect to this server?:");

int numberOfClientsYouNeedToConnect =int.Parse( Console.ReadLine());

for (int i = 0; i < numberOfClientsYouNeedToConnect; i++)

{

Thread newThread = new Thread(new ThreadStart(Listeners));

newThread.Start();

}

}

}

using System;

using System.Net.Sockets;

using System.Threading;

public class Client

{

static public void Main(string[] Args)

{

TcpClient socketForServer;

try

{

socketForServer = new TcpClient("localHost", 10);

}

catch

{

Console.WriteLine(

"Failed to connect to server at {0}:999", "localhost");

return;

}

NetworkStream networkStream = socketForServer.GetStream();

System.IO.StreamReader streamReader =

new System.IO.StreamReader(networkStream);

System.IO.StreamWriter streamWriter =

new System.IO.StreamWriter(networkStream);

Console.WriteLine("\*\*\*\*\*\*\*This is client program who is connected to localhost on port No:10\*\*\*\*\*");

try

{

string outputString;

{

Console.WriteLine("type:");

string str = Console.ReadLine();

while (str != "exit")

{

streamWriter.WriteLine(str);

streamWriter.Flush();

Console.WriteLine("type:");

str = Console.ReadLine();

}

if (str == "exit")

{

streamWriter.WriteLine(str);

streamWriter.Flush();

}

}

}

catch

{

Console.WriteLine("Exception reading from Server");

}

networkStream.Close();

Console.WriteLine("Press any key to exit from client program");

Console.ReadKey();

}

private static string GetData()

{

return "ack";

}

}