Сервер

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Net;

using System.Net.Sockets;

namespace ServerDateEx

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

IPHostEntry iphost = Dns.GetHostEntry("localhost");

IPAddress ipAddr = iphost.AddressList[0];

IPEndPoint ipEndPoint = new IPEndPoint(ipAddr, 11000);

Socket sListener = new Socket(ipAddr.AddressFamily, SocketType.Stream, ProtocolType.Tcp);

try

{

sListener.Bind(ipEndPoint);

sListener.Listen(10);

while (true)

{

MessageBox.Show("Ожидаем соеденение через порт " + ipEndPoint);

Socket handler = sListener.Accept();

string reply = Convert.ToString(DateTime.Now);

byte[] msg = Encoding.UTF8.GetBytes(reply);

handler.Send(msg);

}

}

catch (Exception ex)

{

MessageBox.Show(ex.ToString());

}

}

}

}

Клиент

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Net;

using System.Net.Sockets;

namespace ClientDateEX

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

try

{

textBox1.Text = SendMessageFromSocket(11000);

}

catch (Exception ex)

{

Console.WriteLine(ex.ToString());

}

}

static string SendMessageFromSocket(int port)

{

// Буфер для входящих данных

byte[] bytes = new byte[1024];

// Соединяемся с удаленным устройством

// Устанавливаем удаленную точку для сокета

IPHostEntry ipHost = Dns.GetHostEntry("localhost");

IPAddress ipAddr = ipHost.AddressList[0];

IPEndPoint ipEndPoint = new IPEndPoint(ipAddr, port);

Socket sender = new Socket(ipAddr.AddressFamily, SocketType.Stream, ProtocolType.Tcp);

// Соединяем сокет с удаленной точкой

sender.Connect(ipEndPoint);

string message = "1";

byte[] msg = Encoding.UTF8.GetBytes(message);

// Отправляем данные через сокет

int bytesSent = sender.Send(msg);

// Получаем ответ от сервера

int bytesRec = sender.Receive(bytes);

string text = Encoding.UTF8.GetString(bytes, 0, bytesRec);

return text;

// Используем рекурсию для неоднократного вызова SendMessageFromSocket()

if (message.IndexOf("<TheEnd>") == -1)

SendMessageFromSocket(port);

// Освобождаем сокет

sender.Shutdown(SocketShutdown.Both);

sender.Close();

}

}

}