کد سمت سرور(تبدیل عبارت رشته ای دریافت شده از کلاینت به حروف بزرگ و فرستادن رشته تغییر یافته برای کلاینت)

```csharp
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Net;
using System.Net.Sockets;

namespace mult_srv
{
classProgram
    {
staticvoid Main(string[] args)
        {
TcpListener server = null;
try
            {
int port = 1500;
IPAddress localaddr = IPAddress.Parse("127.0.0.1");
            server = newTcpListener(localaddr, port);
            server.Start();
byte[] bytes = newbyte[1024];
string data = string.Empty;
while (true)
                {
Console.WriteLine("wating for a connection...");
TcpClient client = server.AcceptTcpClient();
Console.WriteLine("connected ");
                data = string.Empty;
NetworkStream stream = client.GetStream();
int i;
while ((i = stream.Read(bytes, 0, bytes.Length)) != 0)
                {
```

```csharp
                    data = System.Text.Encoding.ASCII.GetString(bytes, 0, i);
Console.WriteLine("recived from client: {0}", data);
                    data = data.ToUpper();
byte[] msg = System.Text.Encoding.ASCII.GetBytes(data);
                    stream.Write(msg, 0, msg.Length);
Console.WriteLine("sent to client: {0}", data);
                }
                client.Close();
            }
        }
catch (SocketException err)
        {
Console.WriteLine("error: ", err);
        }
finally
        {
            server.Stop();
        }
Console.Read();
    }
  }
}
```

کد سمت کلاینت(فرستادن عبارت رشته ای برای سرور و دریافت عبارت رشته ای از سرور)

```csharp
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Net;
using System.Net.Sockets;

namespace clientside
{
class Program
    {
static void Main(string[] args)
        {
Socket mysock = new Socket(AddressFamily.InterNetwork, SocketType.Stream, ProtocolType.Tcp);
        mysock.Connect("127.0.0.1", 1500);
while (true)
            {
string str;
            str = Console.ReadLine();
byte[] buffer = Encoding.ASCII.GetBytes(str);
            mysock.Send(buffer);
byte[] cln = new byte[1024];
            mysock.Receive(cln);
string data = Encoding.ASCII.GetString(cln);
Console.WriteLine("message from server : " + data + " ");
            }
        mysock.Close();
        }
    }
}
```

```csharp
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Net;
using System.Net.Sockets;


namespace serverside
{
class Program
    {
static void Main(string[] args)
        {
Console.WriteLine("Server Runing...");
Socket mysock = new Socket(AddressFamily.InterNetwork, SocketType.Stream, ProtocolType.Tcp);
IPEndPoint ipe = new IPEndPoint(IPAddress.Any, 1500);
        mysock.Bind(ipe);
        mysock.Listen(10);
        mysock = mysock.Accept();
while (true)
            {
try
                {

byte[] buffer = new byte[500];
            mysock.Receive(buffer);
string data = Encoding.ASCII.GetString(buffer);
Console.WriteLine("massage from client : " + data + " ");
string str;
            str = Console.ReadLine();
```

```csharp
            byte[] sev = Encoding.ASCII.GetBytes(str);
                    mysock.Send(sev);
                }
        catch (Exception err)
                    {
        Console.WriteLine(err.Message);
                    }
        finally { }
                }
            mysock.Close();
            }
        }
    }
```

```
file:///C:/net-it/mult_srv/mult_srv/bin/Debug/mult_srv.EXE

wating for a connection...
connected
recived from client: this is a test
sent to client: THIS IS A TEST
```

```
file:///C:/net-it/clientside/clientside/bin/Debug/clientside.EXE

this is a test
message from server : THIS IS A TEST
```

```
file:///C:/net-it/serverside/serverside/bin/Debug/serverside.EXE

Server Runing...
massage from client : hi server




hi client
```

```
file:///C:/net-it/clientside/clientside/bin/Debug/clientside.EXE

hi server
message from server : hi client
```