

شروع کار با برنامه نویسی شبکه در C#

سی شارپ و برنامه نویسی شبکه :

همانطور که میدانید دات نت فرم ورک یکی از بزرگترین و بهترین تکنولوژی های موجود در جهان جهت برنامه نویسی میباشد. این فریم ورک در واقع رابطی ما بین ویندوز و زبان برنامه نویسی میباشد که این زبان میتواند سی شارپ ، وی بی و یا هر زبانی که دات نت فرم ورک را پشتیبانی میکند باشد.

برنامه نویسی شبکه یکی از امکانات بزرگ در دات نت فریم ورک میباشد که در فضای نامی زیر قرار دارد :

[System.Net](#)

معرفی فضاهای نامی مربوط به برنامه نویسی شبکه :

[System.Net](#)

این فضای نامی دارای کلاس های متعددی برای برنامه نویسی شبکه در پروتکل های متنوع و گوناگون میباشد از مهمترین کلاس ها میتوان به کلاس های زیر اشاره کرد :

Dns :

برای به دست آوردن آدرس های آی پی ، نام کامپیوتر و غیره به کار میرود.

IPAddress :

برای ذخیره و نگه داری یک آدرس IP

IPEndPoint :

ذخیره و نگه داری آدرس های شبکه به همراه پورت

و یکسری کلاس های دیگر که از حیطة این مقاله خارج است.

[System.Net.Sockets](#)

این فضای نامی نیز از مهم ترین قسمت ها میباشد که در برنامه نویسی شبکه جهت اتصال به سرور یا کلاینت به کار میرود و از کلاس های مهم در این بخش میتوان به کلاس های زیر اشاره کرد :

Socket :

این کلاس یکی از مهمترین های برنامه نویسی شبکه میباشد که وظیفه ارتباط دهی با پروتکل های گوناگون را به استفاده میکنم. ~~تی سی پی سی پی~~ عهده دارد که در این مقاله بنده از پروتکل

TcpListener , TcpClient , UdpClient :

این سری کلاس ها نیز کلاس های کمکی سوکت نامیده میشوند که برای راحتی و خوانایی برنامه به کار میرود که در این مقاله گنجانده نشده اند.

مثال های مربوط به کلاس ها :

برای سهولت در درک کاربرد کلاس های معرفی شده ابتدا به اشاره چند مثال میپردازم :

IPAddress

```
System.Net.IPAddress yourself = System.Net.IPAddress.Any;
MessageBox.Show("Your IP Address : " + yourself.ToString());
System.Net.IPAddress loopback = System.Net.IPAddress.Loopback;
MessageBox.Show("Your LoopBack Address : " + loopback.ToString());
```

EndPoint

```
System.Net.IPAddress ip = System.Net.IPAddress.Parse("196.168.36.30");
System.Net.IPEndPoint ipep = new System.Net.IPEndPoint(
ip, 5050);
MessageBox.Show("Now You can connect to host with IP : "
+ ipep.Address.ToString() + " and Port : "
+ ipep.Port.ToString());
```

Dns

```
MessageBox.Show("Host Name is : " + strHost);
```

Socket(Tcp)

```
System.Net.Sockets.Socket sock =
new System.Net.Sockets.Socket(
System.Net.Sockets.AddressFamily.InterNetwork,
System.Net.Sockets.SocketType.Stream,
System.Net.Sockets.ProtocolType.Tcp);
```

Socket(Udp)

```
System.Net.Sockets.Socket sock =
new System.Net.Sockets.Socket(
System.Net.Sockets.AddressFamily.InterNetwork,
System.Net.Sockets.SocketType.Dgram,
System.Net.Sockets.ProtocolType.Udp);
```

یک برنامه کاربردی کلاینت / سروری :

سرور :

```
using System;
using System.Net;
using System.Net.Sockets;

namespace ServerApp
{
    class ServerSideApp
    {
        static void Main()
        {
            IPAddress ip = IPAddress.Parse("127.0.0.1");
            IPEndPoint ipep = new IPEndPoint(ip, 5050);
            Socket serverSock = new Socket(AddressFamily.InterNetwork,
                SocketType.Stream, ProtocolType.Tcp);
            serverSock.Bind(ipep);
            serverSock.Listen(5);
            Console.WriteLine("Waiting for a client...");
            Socket clientSock = serverSock.Accept();
            string welcome = "Az In Ke Be In Server Vasl Shodid
Mamnun.";
            byte[] data = new byte[50];
            data = System.Text.Encoding.ASCII.GetBytes(welcome);
            clientSock.Send(data);
        }
    }
}
```

در این برنامه ابتدا یک آی پی که در اینجا بنده آی پی محلی را انتخاب کرده ام و سپس یک پورت که 5050 تعریف شده است. کار این برنامه این است که به تمامی درخواست هایی که از سمت کلاینت به پورت 5050 فرستاده میشوند اجازه اتصال داده و یک رشته داده به کلاینت میفرستد.

کلاینت :

```
using System;
using System.Net;
using System.Net.Sockets;

namespace ClientApp
{
    class ClientSideApp
    {
        static void Main()
        {
            IPEndPoint ipep = new
IPPEndPoint(IPAddress.Parse("127.0.0.1"), 5050);
            Socket server = new Socket(AddressFamily.InterNetwork,
                SocketType.Stream, ProtocolType.Tcp);
            try
            {
                server.Connect(ipep);
            }
        }
    }
}
```

```

    }
    catch
    {
        Console.WriteLine("Unable to connect server...");
        return;
    }
    byte[] data = new byte[50];
    int rec = server.Receive(data);
    string strData = System.Text.Encoding.ASCII.GetString(data,
0, rec);
    Console.WriteLine("From Server :: {0} ", strData);
    Console.ReadLine();
    }
}

```

در این برنامه آی پی و پورت تعریف شده در سمت سرور داده شده و با استفاده از دستور کانکت به سرور وصل شده و یک داده از سرور دریافت مینماید.