📔 Server

...

Forenklet læser

Udgivet: 11. september 2023 af [Henrik Vincents Poulsen](#)

```csharp
// A C# Program for Server
using System;
using System.Net;
using System.Net.Sockets;
using System.Text;

namespace Server {

class Program {

// Main Method
static void Main(string[] args)
{
    ExecuteServer();
}
public static void ExecuteServer()
{
    // Establish the local endpoint
    // for the socket. Dns.GetHostName
    // returns the name of the host
    // running the application.
    IPHostEntry ipHost = Dns.GetHostEntry(Dns.GetHostName());
    IPAddress ipAddr = ipHost.AddressList[0];
    IPEndPoint localEndPoint = new IPEndPoint(ipAddr, 11111);
```

```csharp
// Creation TCP/IP Socket using
// Socket Class Constructor
Socket listener = new Socket(ipAddr.AddressFamily,
        SocketType.Stream, ProtocolType.Tcp);

try {

    // Using Bind() method we associate a
    // network address to the Server Socket
    // All client that will connect to this
    // Server Socket must know this network
    // Address
    listener.Bind(localEndPoint);

    // Using Listen() method we create
    // the Client list that will want
    // to connect to Server
    listener.Listen(10);

    while (true) {

        Console.WriteLine("Waiting connection ... ");

        // Suspend while waiting for
        // incoming connection Using
        // Accept() method the server
        // will accept connection of client
        Socket clientSocket = listener.Accept();

        // Data buffer
        byte[] bytes = new Byte[1024];
        string data = null;
```

```csharp
        while (true) {

            int numByte = clientSocket.Receive(bytes);

            data += Encoding.ASCII.GetString(bytes,
                        0, numByte);

            if (data.IndexOf("<EOF>") > -1)
                break;
        }
        Console.WriteLine("Text received -> {0} ", data);
        byte[] message = Encoding.ASCII.GetBytes("Test Server");

        // Send a message to Client
        // using Send() method
        clientSocket.Send(message);

        // Close client Socket using the
        // Close() method. After closing,
        // we can use the closed Socket
        // for a new Client Connection
        clientSocket.Shutdown(SocketShutdown.Both);
        clientSocket.Close();
    }
}

catch (Exception e) {
    Console.WriteLine(e.ToString());
}
}
}
```

```
}
```