

## Упражнение 4

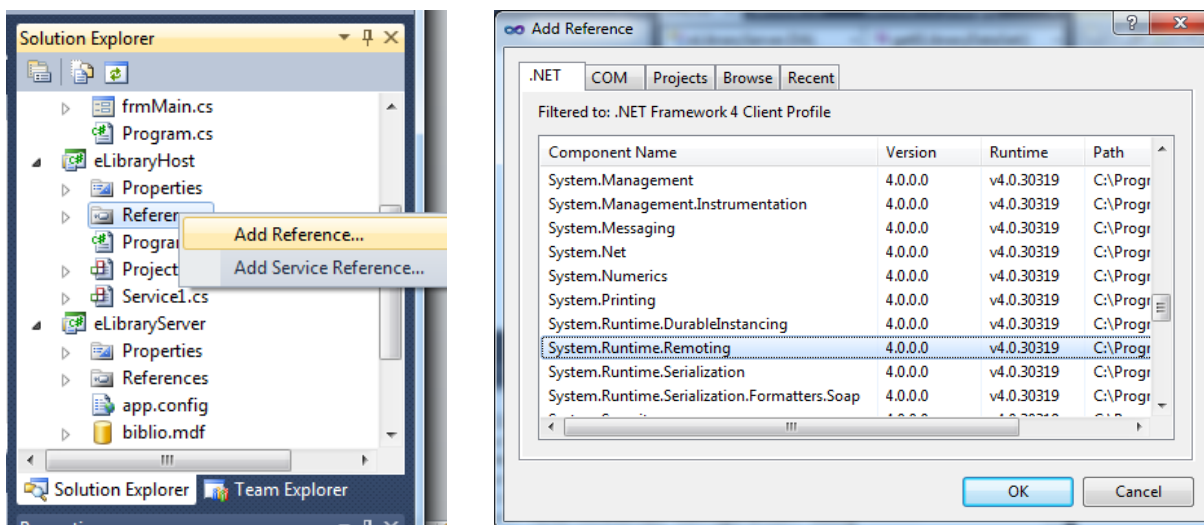
**3.2. От страната на сървъра в проект eLibraryServer: Дефиниране на класа DAL като клас за отдалечен достъп.****Стъпка 1**

В дефиницията на класа DAL добавяме наследяване за *marshal-by-reference*

```
public class DAL : MarshalByRefObject
{
    ...
    public DataSet getELibraryDataSet()
    {
        ...
    }
}
```

**3.3. В проект eLibraryHost: Конфигуриране на отдалечен достъп****Стъпка 1**

- Добавете референция към **System.Runtime.Remoting** (от таб “.NET”).



- Добавете референция към проекта eLibraryServer от таб Projects.
- Добавете референция към проекта eLibraryClient от таб Projects.

**Стъпка 2**

Редактирайте Service1.cs :

- Добавете библиотеките:
 

```
using System.Runtime.Remoting;
using System.Runtime.Remoting.Channels;
using System.Runtime.Remoting.Channels.Tcp;
```

- Създайте функция **doRemotingConfiguration()** с командите за конфигуриране на отдалечен достъп.
- Извикайте *doRemotingConfiguration* във функцията *OnStart()*.

```
using System.Runtime.Remoting;
using System.Runtime.Remoting.Channels;
using System.Runtime.Remoting.Channels.Tcp;

public partial class Service1 : ServiceBase
{
    protected override void OnStart(string[] args)
    {
        ...
        doRemotingConfiguration();
    }
    private void doRemotingConfiguration()
    {
        TcpServerChannel channel;
        //-- create and register a TCP server channel on port 8089
        channel = new TcpServerChannel(8089);
        ChannelServices.RegisterChannel(channel, false);
        //-- register the service for remote access in SingleCall mode
        RemotingConfiguration.RegisterWellKnownServiceType(typeof(eLibraryServer.DAL), "eLibraryServer.rem", WellKnownObjectMode.SingleCall);
    }
}
```

### 3. 4. От страната на клиента: Съвързване с отдалечен обект

#### Стъпка 1

- Добавете референция към **System.Runtime.Remoting** (от таб “.NET”)

#### Стъпка 2

Във формата **frmMain** :

- Добавете библиотеките:
 

```
using System.Runtime.Remoting;
using System.Runtime.Remoting.Channels;
using System.Runtime.Remoting.Channels.Tcp;
```
- Създайте функция **doClientRemotingConfiguration()** с командите за конфигуриране на отдалечен достъп.
- Извикайте *doClientRemotingConfiguration* във функцията *frmMain\_Load()*

```
using System.Runtime.Remoting;
using System.Runtime.Remoting.Channels;
using System.Runtime.Remoting.Channels.Tcp;

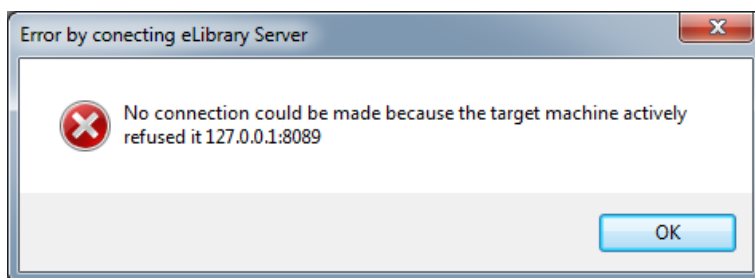
public partial class frmMain : Form
{
    ...
    private void frmMain_Load(object sender, EventArgs e)
    {
        ...
        doClientRemotingConfiguration();
    }
}
```

```
private void doClientRemotingConfiguration()
{
    TcpClientChannel channel= new TcpClientChannel();
    //-- Register a TCP client channel

    ChannelServices.RegisterChannel(channel, false);
    //-- Register the remote class
    RemotingConfiguration.RegisterWellKnownClientType(typeof(eLibraryServer.DAL), "tcp://localhost:8089/eLibraryServer.rem");
}
```

### Стъпка 3

- За да се уверим, че приложението, след направените промени, вече се обръща към отдалечения сървис, спираме сървиса **1eLibraryHost** и стартираме клиентското приложение. Тъй като отдалеченият сървис не е активен, трябва да се появи следното съобщение:



- Стартирайте сървиса **1eLibraryHost** от Server Explorer на Visual Studio. Стартирайте клиентското приложение. Сега връзката трябва да се осъществи успешно, и при активиране на **show books**, екранът изглежда по същия начин, като при използване на локална връзка с *eLibraryServer.dll*, само, че използва вече отдалечена връзка чрез създадения сървис.

