WCF Assignment Guide

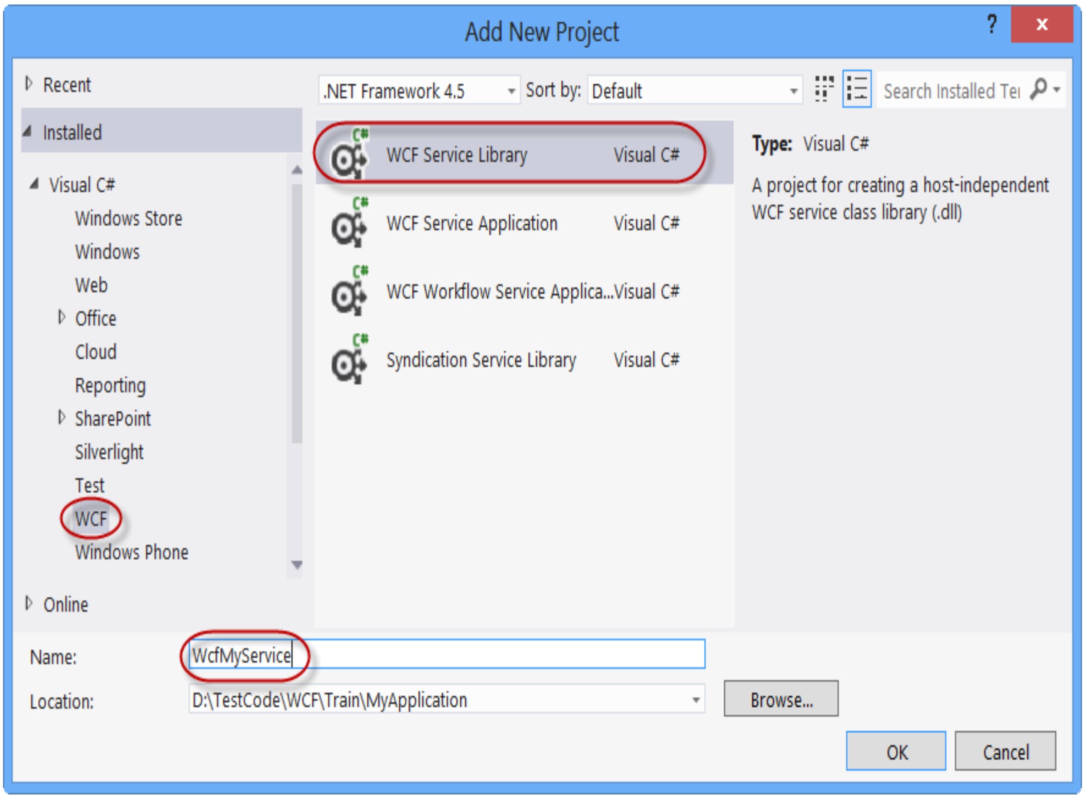
❖ **Introduction** : Developing a Client-Server application with WCF and config multiple-endpoint

❖ **Requirement** :

- Open Visual Studio 2012 (.NET Framework 4.5) , select File | New Project |Other Project Type|Visual Studio Solutions|Blank Solution and named "MyApplication"

➢ Step 1 : Create Remote Object by WCF Service

1.1 From menu File|Add New Project, In WCF | WCF Service Library and named WcfMyService , click OK to finish



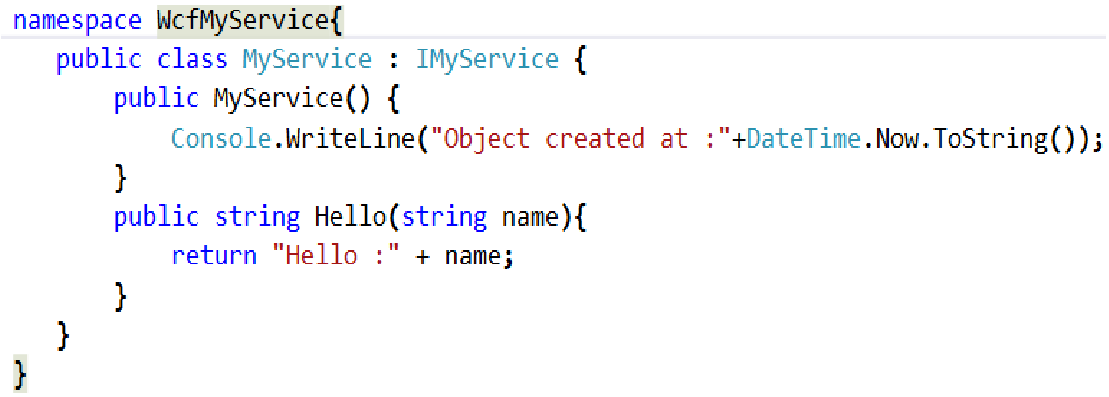
Name: WcfMyService

Location: D:\TestCode\WCF\Train\MyApplication

- Rename IService1.cs to IMyService.cs , Service.cs and to MyService.cs
- Write codes in IMyService.cs as the following :

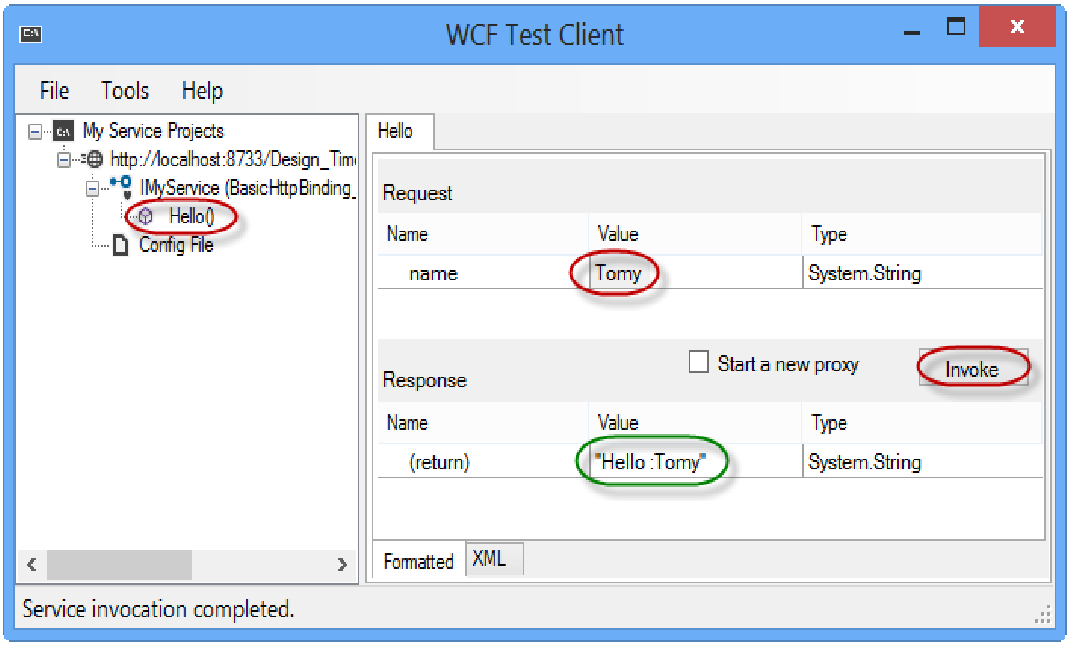
```
namespace WcfMyService
{
    public interface IMyService
    {
        [OperationContract]
        string Hello(string name);
    }

}
```

- Write codes in MyService.cs as the following :



```
namespace WcfMyService{
    public class MyService : IMyService {
        public MyService() {
            Console.WriteLine("Object created at :"+DateTime.Now.ToString());
        }
        public string Hello(string name){
            return "Hello :" + name;
        }
    }
}
```

1.2 Press Ctrl+F5 to test wcf application . When “WCF Test Client” dialog showed , click on “Hello” method and input value for parameter | Click **Invoke** to execute this method.



Request

Name	Value	Type
name	Tomy	System.String

□ Start a new proxy

Response

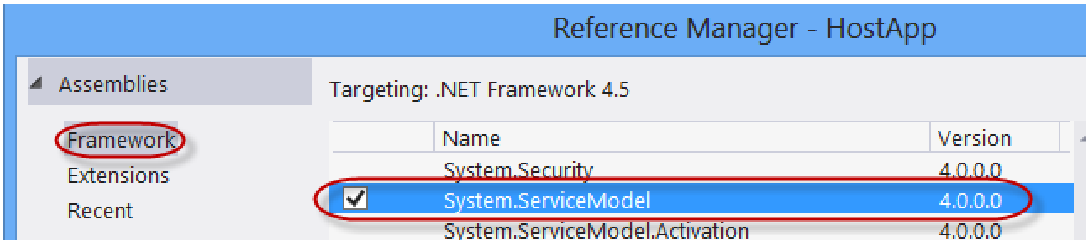
Name	Value	Type
(return)	"Hello :Tomy"	System.String

Service invocation completed.

Step 2 : Create a Host Application

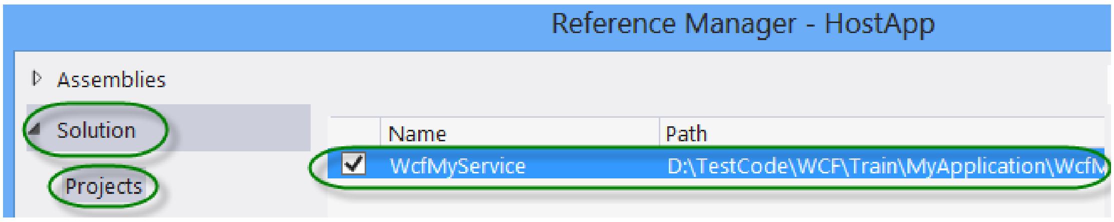
2.1 From menu File|Add New Project, In Project Type | Console Application and named ConsoleHostApp

2.2 .On this project, right-click | Add Reference , add reference to WcfMyService.dll and System.ServiceModel.dll



Targeting: .NET Framework 4.5

	Name	Version
	System.Security	4.0.0.0
☑	System.ServiceModel	4.0.0.0
	System.ServiceModel.Activation	4.0.0.0



Reference Manager - HostApp

▷ Assemblies

◢ Solution

Projects

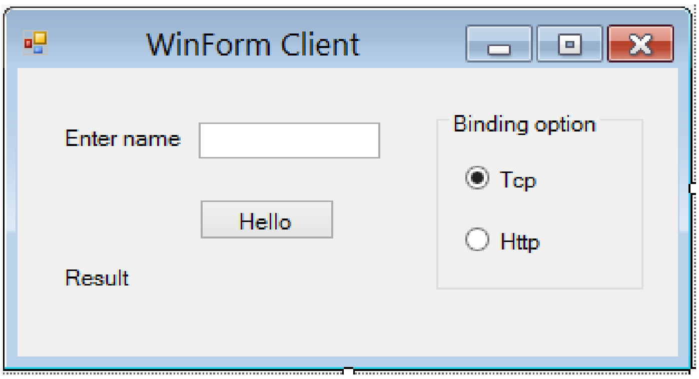
	Name	Path
☑	WcfMyService	D:\TestCode\WCF\Train\MyApplication\WcfM

2.3. Write code in Main method as the following :

```
using System.ServiceModel;
using WcfMyService;
namespace ConsoleHostApp
{
   class Program
   {
        static void Main(string[] args)
        {
            ServiceHost host = new ServiceHost(typeof(MyService));
            host.AddServiceEndpoint(typeof(IMyService), new NetTcpBinding(),
             "net.tcp://localhost:8989/MyService");
            host.AddServiceEndpoint(typeof(IMyService), new WSHttpBinding(),
                "http://localhost:8888/MyService");
            host.Open();
            Console.WriteLine("Server started....");
            Console.ReadLine();
        }
   }
}
```

- **Step 3 : Create a Client Application**

3.1 From menu File|Add New Project, In Project Type | Windows Forms Application and named ClientApp. On this project, right-click | Add Reference , add reference to WcfMyService.dll and System.ServiceModel.dl . Form have user interface as the table below :



WinForm Client

Enter name

Hello

Result

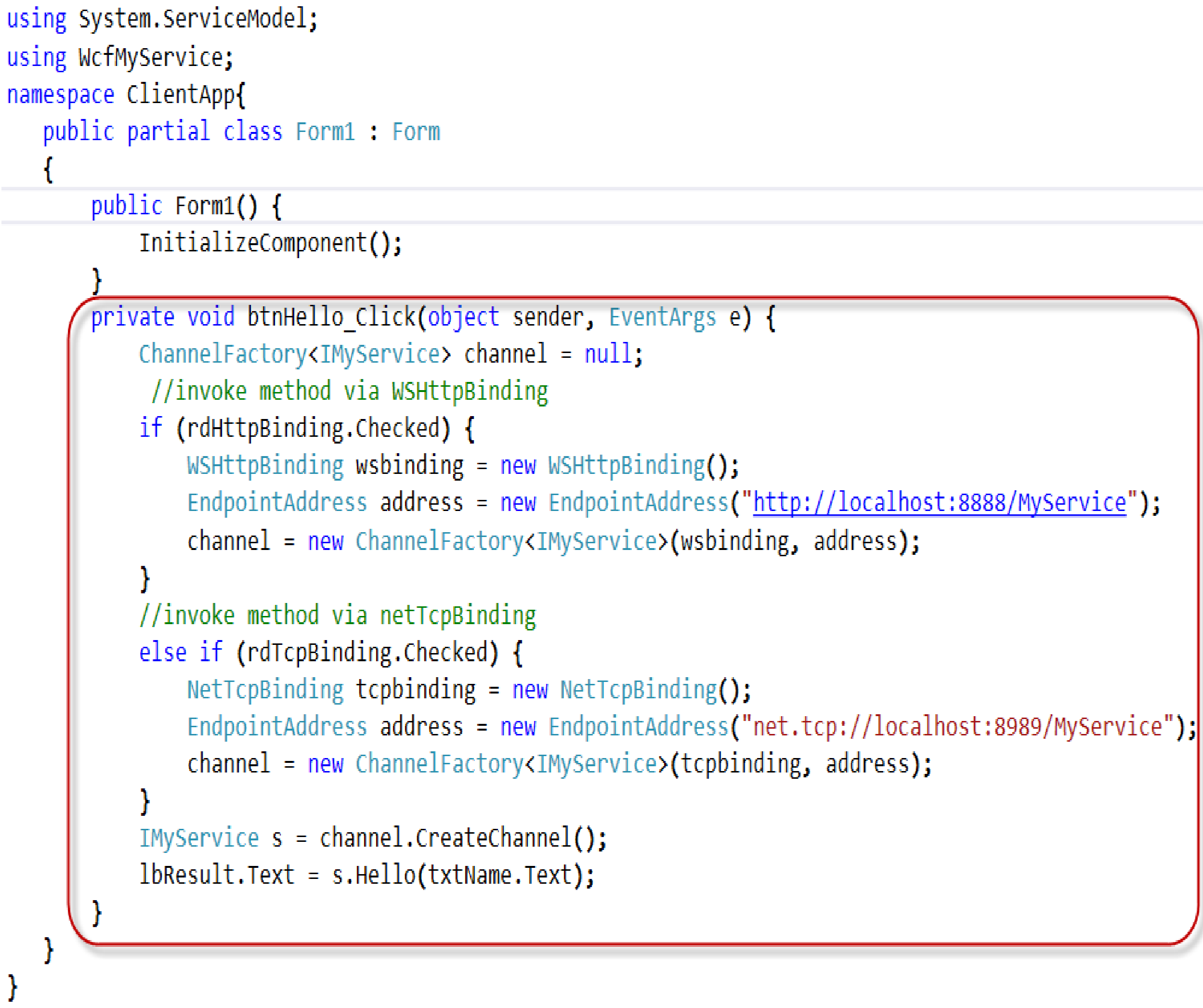
Binding option

● Tcp

○ Http

Control Type	Properties/ Event	Value
TextBox	Name	txtName
Label	Text	Enter name
GroupBox	Text	Binding option
Label	Name	lbResult
	Text	Result
Button	Name	btnHello
	Text	Hello
	Event: Click	btnHello_Click
RadioButton	Name	rdHttpBinding
	Text	Http
RadioButton	Name	rdTcpBinding
	Text	Tcp
	Checked	True

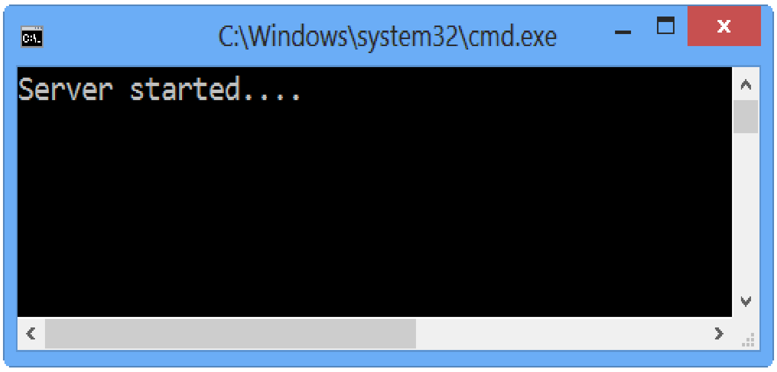
- Double-click on "Hello" button and write codes as the following :



```
using System.ServiceModel;
using WcfMyService;
namespace ClientApp{
   public partial class Form1 : Form
   {
       public Form1() {
           InitializeComponent();
       }
       private void btnHello_Click(object sender, EventArgs e) {
           ChannelFactory<IMyService> channel = null;
            //invoke method via WSHttpBinding
           if (rdHttpBinding.Checked) {
               WSHttpBinding wsbinding = new WSHttpBinding();
               EndpointAddress address = new EndpointAddress("http://localhost:8888/MyService");
               channel = new ChannelFactory<IMyService>(wsbinding, address);
           }
           //invoke method via netTcpBinding
           else if (rdTcpBinding.Checked) {
               NetTcpBinding tcpbinding = new NetTcpBinding();
               EndpointAddress address = new EndpointAddress("net.tcp://localhost:8989/MyService");
               channel = new ChannelFactory<IMyService>(tcpbinding, address);
           }
           IMyService s = channel.CreateChannel();
           lbResult.Text = s.Hello(txtName.Text);
       }
   }
}
```

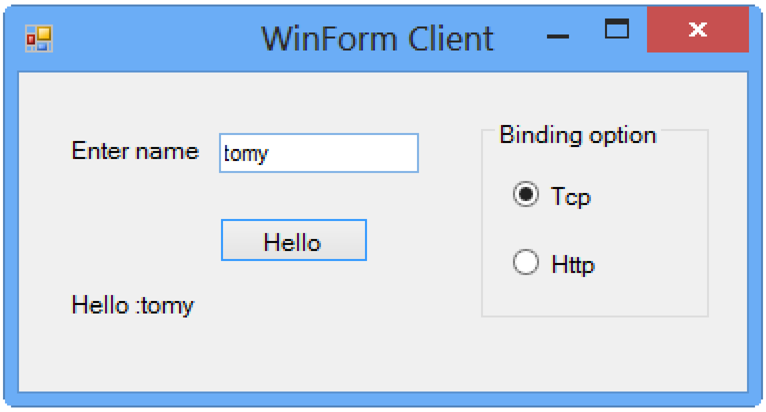
- **Step 4 : Run applications**

4.1 Run ConsoleHostApp



```
Server started....
```

4.2 Run ClientApplication



Enter name tomy

Hello :tomy

● Tcp

○ Http