Practical no 2

AIM: Write a program to implement the operation can receive request and will return a response in two ways. a) One - Way operation b) Request –Response

A)One-Way

1. WebForm.aspx file

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WebForm1.aspx.cs" Inher-
its="prac2a.WebForm1" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
    <asp:Button ID="Button1" runat="server" onclick="Button1_Click" Text="Button" />
    <br />
    <asp:Label ID="lblPagedate" runat="server">  </asp:Label>
    <br />
<br />
     <asp:Label ID="lblServicedate" runat="server">  </asp:Label>
    </div>
    </form>
</body>
</html>
```

2. WebForm.aspx.cs file

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace prac2a
{
    public partial class WebForm1 : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {

        }
        protected void Button1_Click(object sender, EventArgs e)
```

```
        {
            //Time show before the service is calling
            lblPagedate.Text = "on the load time the time is " + DateTime.Now.ToString();
            ServiceReference1.Service1Client sc = new ServiceReference1.Service1Client();
            sc.OneWayMessage();
            //after service is calling that time show
            lblServicedate.Text = "After Calling the service the time is " +
DateTime.Now.ToString();
        }
    }
}
```

3. IService.cs:-

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Runtime.Serialization;
using System.ServiceModel;
using System.ServiceModel.Web;
using System.Text;

namespace prac2a
{
    [ServiceContract]
    public interface IService1
    {
        [OperationContract(IsOneWay = true)]
        void OneWayMessage();
    }
}
```
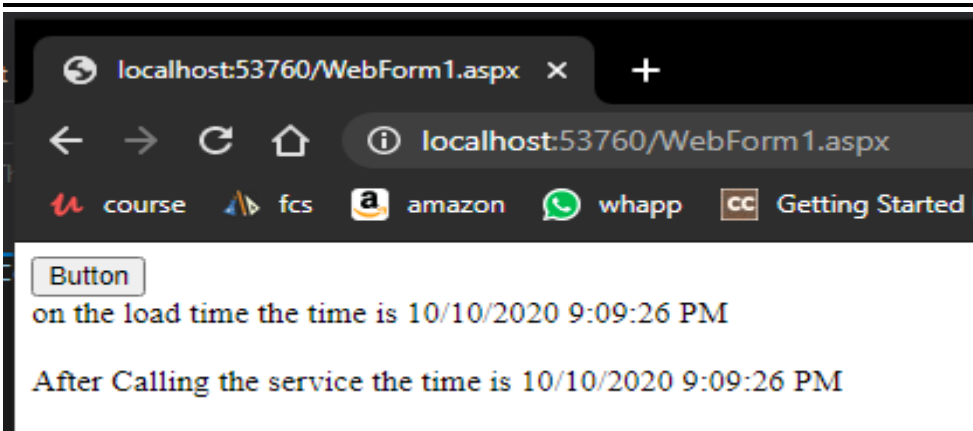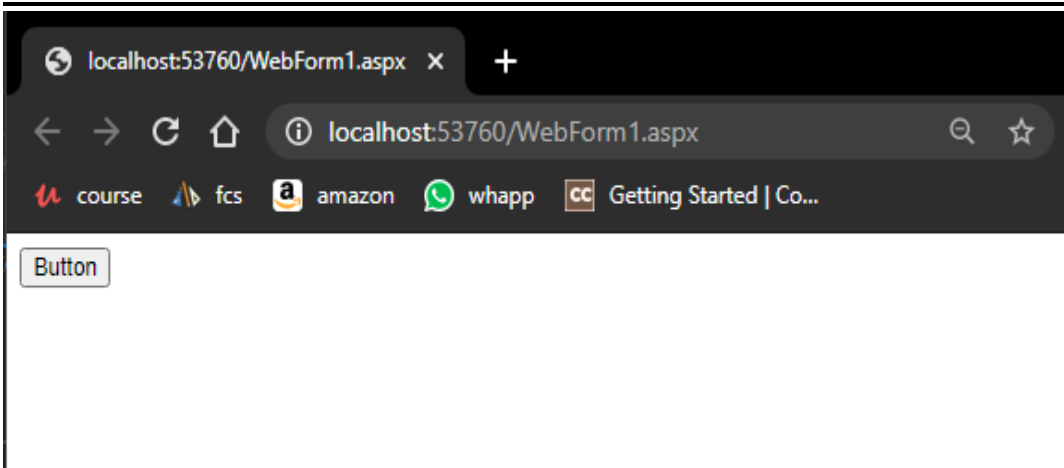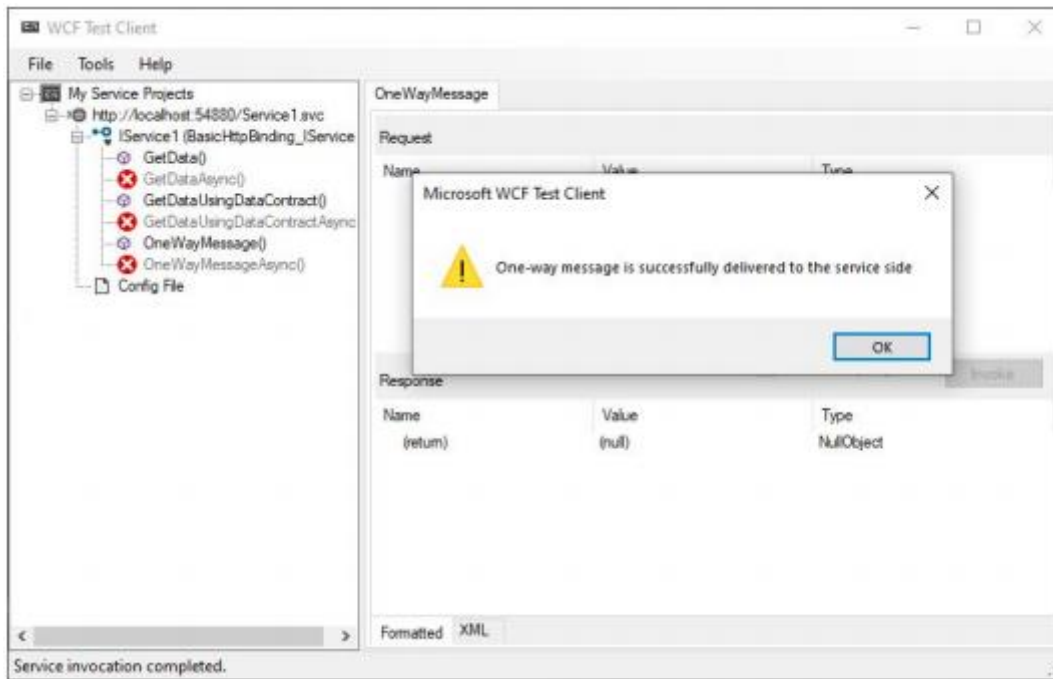
4. Service1.svc.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Runtime.Serialization;
using System.ServiceModel;
using System.ServiceModel.Web;
using System.Text;
using System.Threading;

namespace prac2a
{
    public class Service1 : IService1
    {
        public void OneWayMessage()
        {
            Thread.Sleep(2000);
        }
    }}
```

Output:-

WCF Test Client

File   Tools   Help

My Service Projects
  http://localhost:54880/Service1.svc
    IService1 (BasicHttpBinding_IService
      GetData()
      GetDataAsync()
      GetDataUsingDataContract()
      GetDataUsingDataContractAsync
      OneWayMessage()
      OneWayMessageAsync()
    Config File

OneWayMessage

Request

Name	Value	Type

Microsoft WCF Test Client   ✕

⚠ One-way message is successfully delivered to the service side

OK

Response

Name	Value	Type
(return)	(null)	NullObject

Formatted XML

Service invocation completed.

localhost:53760/WebForm1.aspx   ✕   +

← → C ⌂ ⓘ localhost:53760/WebForm1.aspx   🔍 ☆

course   fcs   amazon   whapp   Getting Started | Co...

Button

localhost:53760/WebForm1.aspx   ✕   +

← → C ⌂ ⓘ localhost:53760/WebForm1.aspx

course   fcs   amazon   whapp   Getting Started |

Button

on the load time the time is 10/10/2020 9:09:26 PM

After Calling the service the time is 10/10/2020 9:09:26 PM

B) Request –Response

1. WebForm.aspx file

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WebForm1.aspx.cs" Inherits="Wcf-
Service2.WebForm1" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <asp:Button ID="Button1" runat="server" onclick="Button1_Click" Text="Button" />
    <br />
    <asp:Label ID="lblPagedate" runat="server"></asp:Label>
    <br />
<br />
    <asp:Label ID="lblServicedate" runat="server"></asp:Label>

        </div>
    </form>
</body>
</html>
```

2. WebForm.aspx.cs file

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
namespace WcfService2
{
    public partial class WebForm1 : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {

        }
        protected void Button1_Click(object sender, EventArgs e)
        {
            lblPagedate.Text = "Page date is " + DateTime.Now.ToString();
            ServiceReference1.Service1Client sc = new ServiceReference1.Service1Client();
            lblServicedate.Text = sc.RequestReplyPattern();
        }
    }
}
```

Web Services	T.Y.B.S.c.C.S.Sem-V

3. IService.cs:-

```csharp
using System;
using System.Collections.Generic;
using System.Linq;
using System.Runtime.Serialization;
using System.ServiceModel;
using System.ServiceModel.Web;
using System.Text;
namespace WcfService2
{
    [ServiceContract]
    public interface IService1
    {
        [OperationContract(IsOneWay = false)]
        //declare the method which return type is string
        string RequestReplyPattern();

    }
}
```

4. Service1.svc.cs

```csharp
using System;
using System.Collections.Generic;
using System.Linq;
using System.Runtime.Serialization;
using System.ServiceModel;
using System.ServiceModel.Web;
using System.Text;
using System.Threading;

namespace WcfService2
{
    public class Service1 : IService1
    {
        public string RequestReplyPattern()
        {
            Thread.Sleep(2000);
            return "current time of Service is" + DateTime.Now.ToString();
        }
    }
}
```

Output:-