

Practical No: 6.1

Aim: Create A Simple XML Web Service Application

Source Code:

File -> New -> Website -> APS.NET Empty Web Site (WebService)

Add->New Item->Web Service

WebService.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Services;

[WebService(Namespace = "http://tempuri.org/")]
[WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
public class WebService : System.Web.Services.WebService {

    public WebService ()
    {
    }

    [WebMethod]
    public int Add(int a,int b)
    {
        return a + b;
    }
}
```

WebService.cs

Copy the link (http://localhost:53702/WebService.asmx)

WebService Web Service x +

localhost:53702/WebService.asmx

WebService

The following operations are supported. For a formal definition, please review the [Service Description](#).

- [Add](#)

This web service is using <http://tempuri.org/> as its default namespace.

Recommendation: Change the default namespace before the XML Web service is made public.

Each XML Web service needs a unique namespace in order for client applications to distinguish it from other services on the Web. <http://tempuri.org/> is available for XML Web services that are under development, but published XML Web services should use a more permanent namespace.

Your XML Web service should be identified by a namespace that you control. For example, you can use your company's Internet domain name as part of the namespace. Although many XML Web service namespaces look like URLs, they need not point to actual resources on the Web. (XML Web service namespaces are URIs.)

For XML Web services created using ASP.NET, the default namespace can be changed using the `WebService` attribute's `Namespace` property. The `WebService` attribute is an attribute applied to the class that contains the XML Web service methods. Below is a code example that sets the namespace to "http://microsoft.com/webservices/":

C#

```
[WebService(Namespace="http://microsoft.com/webservices/")]
public class MyWebService {
    // implementation
}
```

Visual Basic

```
<WebService(Namespace="http://microsoft.com/webservices/")> Public Class MyWebService
    ' implementation
End Class
```

C++

```
[WebService(Namespace="http://microsoft.com/webservices/")]
public ref class MyWebService {
    // implementation
};
```

For more details on XML namespaces, see the W3C recommendation on [Namespaces in XML](#).

For more details on WSDL, see the [WSDL Specification](#).

For more details on URIs, see [RFC 2396](#).

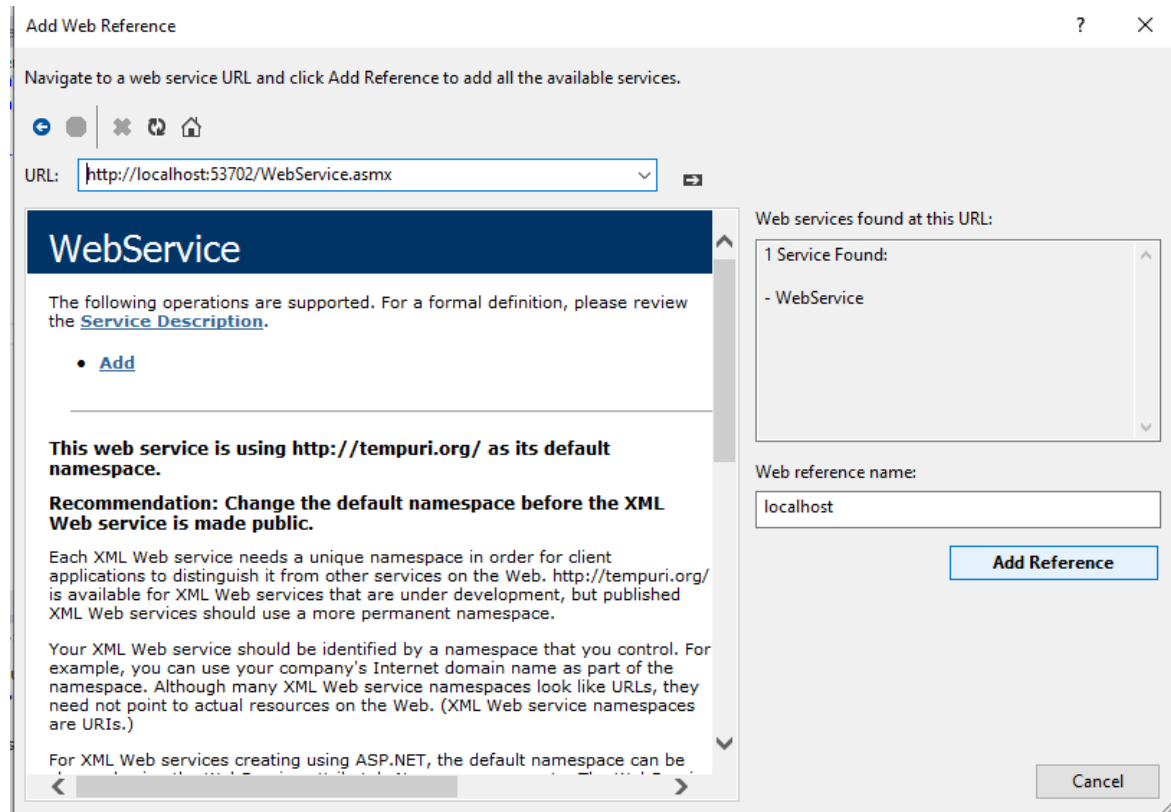
Type here to search

21:27 09-04-2019

File -> New -> Website -> APS.NET Empty Web Site (WebClient)

Add Web Service Reference

Add -> Add Service Reference -> Advanced -> Add Web Reference -> Paste the link -> Add Reference



Add -> New -> web Form (WebClient)

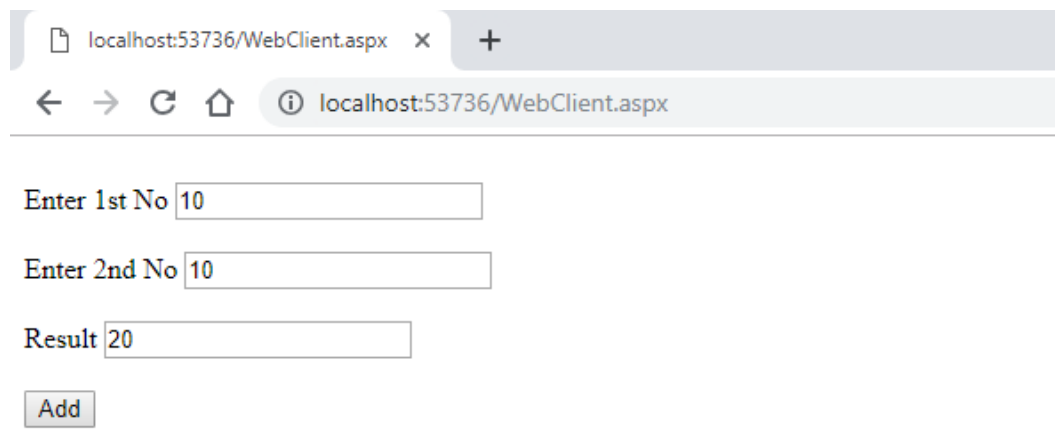
WebClient.aspx.cs

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

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

    protected void B1_Click(object sender, EventArgs e)
    {
        localhost.WebService wc = new localhost.WebService();
        int a = int.Parse(T1.Text);
        int b = int.Parse(T2.Text);
        int c = wc.Add(a, b);
        T3.Text = c.ToString();
    }
}
```

Output



The screenshot shows a web browser window with a single tab titled 'localhost:53736/WebClient.aspx'. The address bar also displays 'localhost:53736/WebClient.aspx'. The page content includes three input fields with labels: 'Enter 1st No' with the value '10', 'Enter 2nd No' with the value '10', and 'Result' with the value '20'. Below these fields is a button labeled 'Add'.

localhost:53736/WebClient.aspx × +

← → ↻ 🏠 ⓘ localhost:53736/WebClient.aspx

Enter 1st No

Enter 2nd No

Result

Practical No: 6.2

Aim: Create A Simple XML Web Service Application using Database

Source Code:

File -> New -> Website -> APS.NET Empty Web Site (WebService1)

Add -> New Item -> Web Service1

Add a Database

Add -> New Item -> SQL Server Database (Employee.mdf)

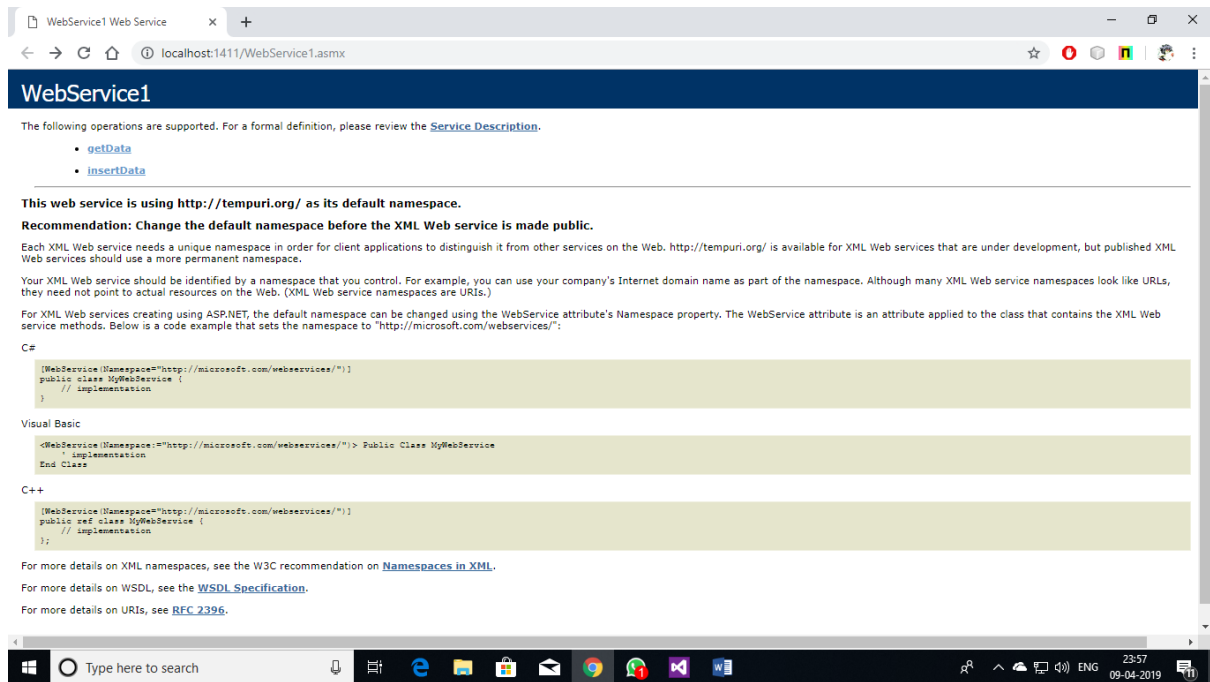
WebService.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Services;
using System.Data;
using System.Data.SqlClient;
using System.Web.Configuration;

[WebService(Namespace = "http://tempuri.org/")]
[WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
public class WebService1 : System.Web.Services.WebService
{
    SqlConnection cn;
    SqlCommand cm;
    SqlDataAdapter da;
    DataSet ds = new DataSet();
    public WebService1 ()
    {
    }

    [WebMethod]
    public DataSet getData()
    {
        cn = new
SqlConnection(WebConfigurationManager.ConnectionStrings["con"].ConnectionString);
        cm = new SqlCommand("select * from emp", cn);
        da = new SqlDataAdapter(cm);
        da.Fill(ds, "empl");
        return (ds);
    }

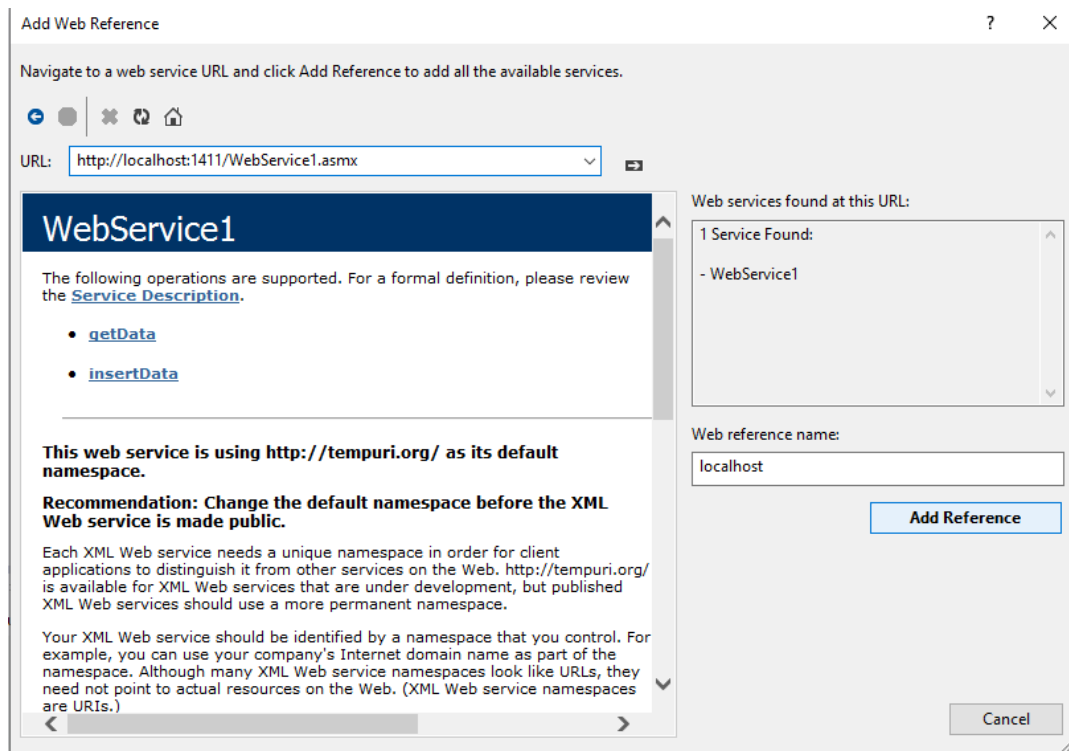
    [WebMethod]
    public void insertData(int id, string nm, int sal)
    {
        cn = new
SqlConnection(WebConfigurationManager.ConnectionStrings["con"].ConnectionString);
        cn.Open();
        cm = new SqlCommand("insert into emp values(@eid,@name,@salary)", cn);
        cm.Parameters.AddWithValue("eid", id);
        cm.Parameters.AddWithValue("name", nm);
        cm.Parameters.AddWithValue("salary", sal);
        cm.ExecuteNonQuery();
        cn.Close();
    }
}
```



File -> New -> Website -> APS.NET Empty Web Site (WebClient1)

Add Web Service Reference

Add -> Add Service Reference -> Advanced -> Add Web Reference -> Paste the link -> Add Reference



Add-> New->web Form(WebClient1)

WebClient1.aspx.cs

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

public partial class WebClient1 : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        localhost.WebService1 ws = new localhost.WebService1();
        DataSet ds = new DataSet();
        ds = ws.getData();
        GridView1.DataSource = ds;
        GridView1.DataBind();
    }
    protected void Insert_Click(object sender, EventArgs e)
    {
        localhost.WebService1 ws = new localhost.WebService1();
        ws.insertData(Convert.ToInt32(t1.Text), t2.Text, Convert.ToInt32(t3.Text));
        l4.Text = "Data Inserted !!!!!";
    }
}
```

Output

The first screenshot shows the initial state of the web application. It features a table with three rows of data: Harry (100000), Hermione (150000), and Ron (80000). Below the table are three input fields labeled 'Id', 'Name', and 'Salary', and an 'Insert' button.

Id	Name	Salary
1	Harry	100000
2	Hermione	150000
3	Ron	80000

Id
Name
Salary

The second screenshot shows the application after the 'Insert' button has been clicked. The 'Id' field now contains the value '4', and the 'Name' field contains 'DumbleDore'. The 'Salary' field is empty. The 'Insert' button now displays the text 'Data Inserted !!!!!'.

Id	Name	Salary
1	Harry	100000
2	Hermione	150000
3	Ron	80000

Id 4
Name DumbleDore
Salary

The third screenshot shows the application after the 'Insert' button has been clicked and the page has been refreshed. The table now displays four rows of data, including the newly added entry: Harry (100000), Hermione (150000), Ron (80000), and DumbleDore (30000). The input fields and the 'Insert' button are in their initial state.

Id	Name	Salary
1	Harry	100000
2	Hermione	150000
3	Ron	80000
4	DumbleDore	30000

Id
Name
Salary

Practical No: 6.3

Aim: Create A Simple WCF Web Service Application

Source Code:

File -> New Project -> WCF Service Application

IService1.cs

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

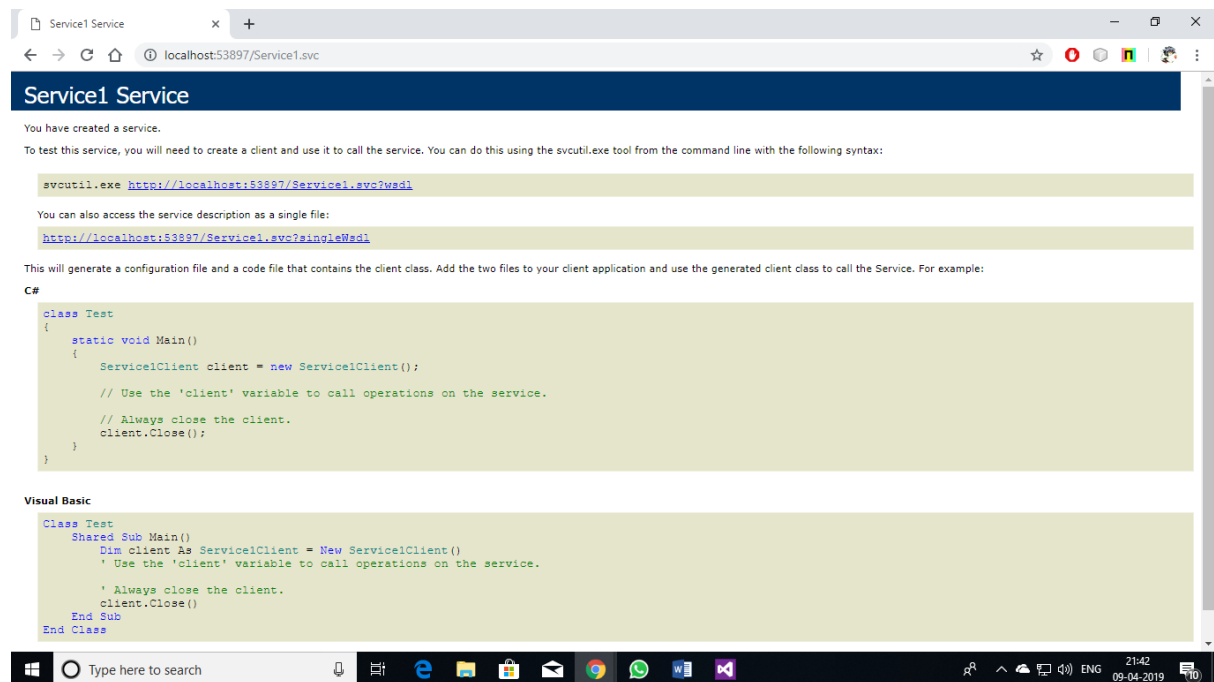
namespace WcfService
{
    [ServiceContract]
    public interface IService1
    {
        [OperationContract]
        int add(int a, int b);
    }
}
```

Service1.svc

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

namespace WcfService
{
    public class Service1 : IService1
    {
        public int add(int a,int b)
        {
            return a+b;
        }
    }
}
```

Run ***IService1.cs*** and copy the link(<http://localhost:53897/Service1.svc>)



Service1 Service

You have created a service.

To test this service, you will need to create a client and use it to call the service. You can do this using the svcutil.exe tool from the command line with the following syntax:

```
svcutil.exe http://localhost:53897/Service1.svc?wsdl
```

You can also access the service description as a single file:

```
http://localhost:53897/Service1.svc?singleWsdl
```

This will generate a configuration file and a code file that contains the client class. Add the two files to your client application and use the generated client class to call the Service. For example:

C#

```
class Test
{
    static void Main()
    {
        Service1Client client = new Service1Client();

        // Use the 'client' variable to call operations on the service.

        // Always close the client.
        client.Close();
    }
}
```

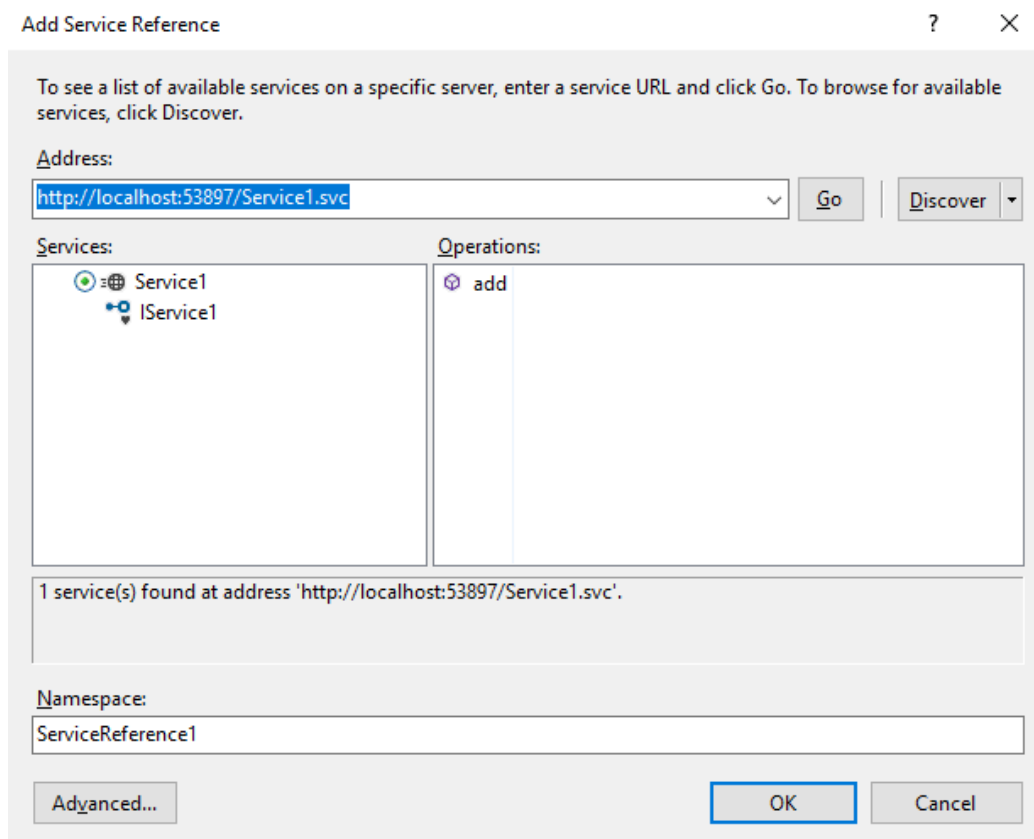
Visual Basic

```
Class Test
Shared Sub Main()
    Dim client As Service1Client = New Service1Client()
    ' Use the 'client' variable to call operations on the service.

    ' Always close the client.
    client.Close()
End Sub
End Class
```

File -> New Web Site -> ASP.NET Empty Web Site (WebClient3)

Add -> Add Service Reference



Add Service Reference

To see a list of available services on a specific server, enter a service URL and click Go. To browse for available services, click Discover.

Address:

<http://localhost:53897/Service1.svc> **Go** **Discover**

Services:

- Service1
- IService1

Operations:

- add

1 service(s) found at address 'http://localhost:53897/Service1.svc'.

Namespace:

ServiceReference1

Advanced... **OK** **Cancel**

Add-> New->web Form(WebClient3)

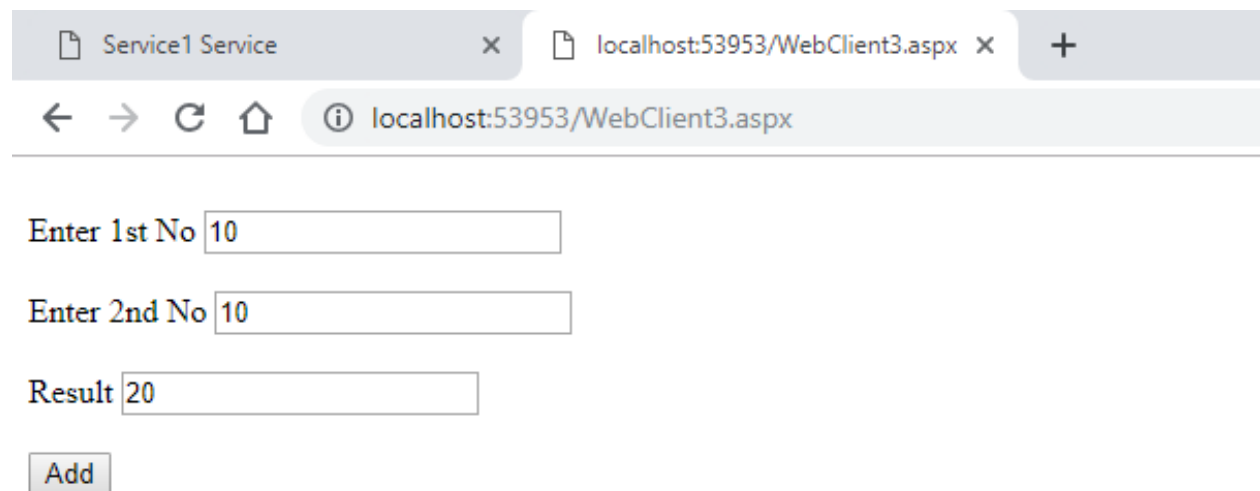
WebClient3.aspx.cs

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

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

    }
    protected void B1_Click(object sender, EventArgs e)
    {
        ServiceReference1.Service1Client sc = new ServiceReference1.Service1Client();
        int a = int.Parse(T1.Text);
        int b = int.Parse(T2.Text);
        int c = sc.add(a, b);
        T3.Text = c.ToString();
    }
}
```

Output



Service1 Service x localhost:53953/WebClient3.aspx x +

← → ↻ 🏠 ⓘ localhost:53953/WebClient3.aspx

Enter 1st No

Enter 2nd No

Result

Practical No: 6.4

Aim: Create A Simple WCF Web Service Application using Database

Source Code:

File -> New Project -> WCF Service Application

IService1.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.Data;

namespace WcfService1
{
    [ServiceContract]
    public interface IService1
    {
        [OperationContract]
        DataSet getData();

        [OperationContract]
        void insertData();

        [OperationContract]
        CompositeType GetDataUsingDataContract(CompositeType composite);
    }
    [DataContract]
    public class CompositeType
    {
        int id, sal;
        string nm;

        [DataMember]
        public int Id
        {
            get { return id; }
            set { id = value; }
        }
        [DataMember]
        public int Sal
        {
            get { return sal; }
            set { sal = value; }
        }
        [DataMember]
        public string Name
        {
            get { return nm; }
            set { nm = value; }
        }
    }
}
```

Add a Database

Add -> New Item -> SQL Server Database (Employee.mdf)

Service1.svc

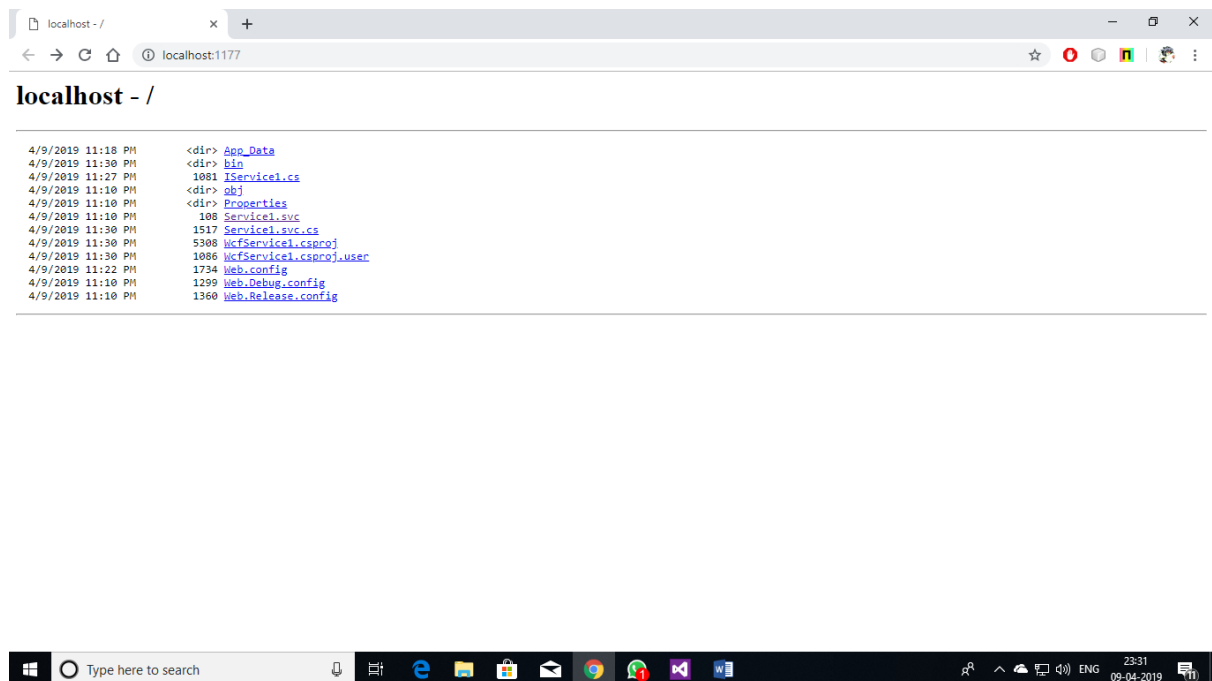
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Runtime.Serialization;
using System.ServiceModel;
using System.ServiceModel.Web;
using System.Text;
using System.Data;
using System.Data.SqlClient;
using System.Web.Configuration;

namespace WcfService1
{
    // NOTE: You can use the "Rename" command on the "Refactor" menu to change the
    class name "Service1" in code, svc and config file together.
    // NOTE: In order to launch WCF Test Client for testing this service, please
    select Service1.svc or Service1.svc.cs at the Solution Explorer and start debugging.
    public class Service1 : IService1
    {
        SqlConnection cn;
        SqlCommand cm;
        SqlDataAdapter da;
        DataSet ds = new DataSet();

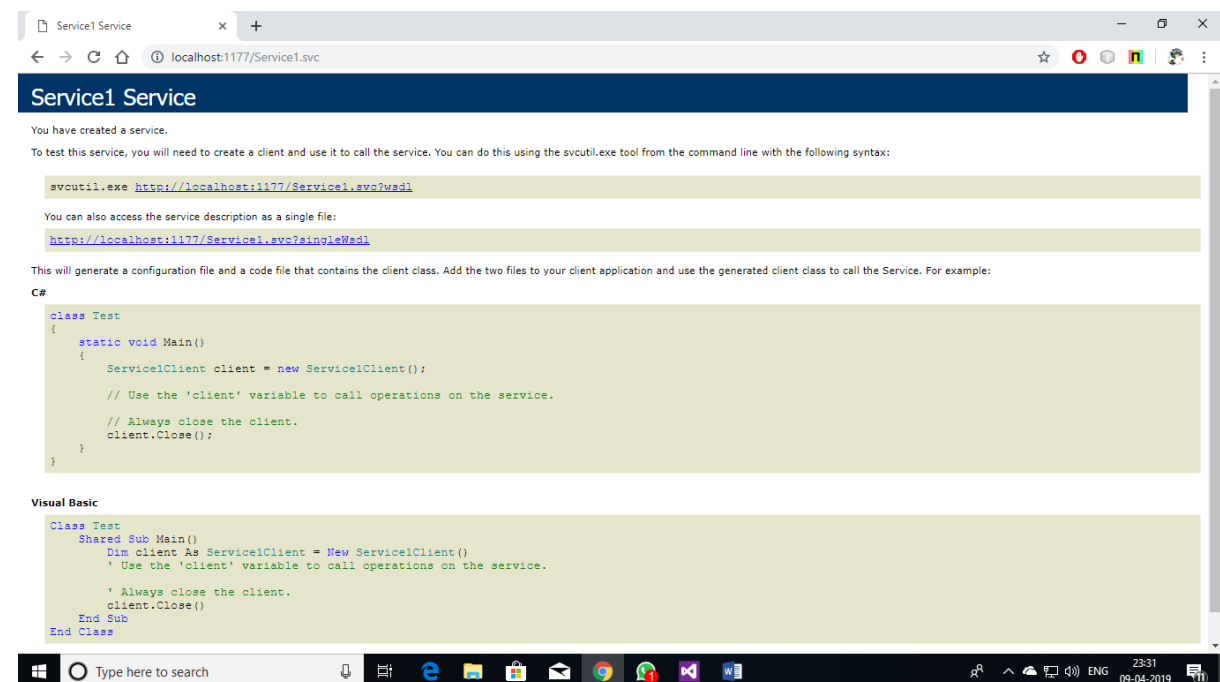
        public DataSet getData()
        {
            cn = new
SqlConnection(WebConfigurationManager.ConnectionStrings["con"].ConnectionString);
            cm = new SqlCommand("select * from emp", cn);
            da = new SqlDataAdapter(cm);
            da.Fill(ds, "empl");
            return(ds);
        }

        public void insertData(int id,string nm,int sal)
        {
            cn = new
SqlConnection(WebConfigurationManager.ConnectionStrings["con"].ConnectionString);
            cn.Open();
            cm = new SqlCommand("insert into emp values(@eid,@name,@salary)",cn);
            cm.Parameters.AddWithValue("eid", id);
            cm.Parameters.AddWithValue("name", nm);
            cm.Parameters.AddWithValue("salary", sal);
            cm.ExecuteNonQuery();
            cn.Close();
        }
    }
}
```

Run *IService1.cs*

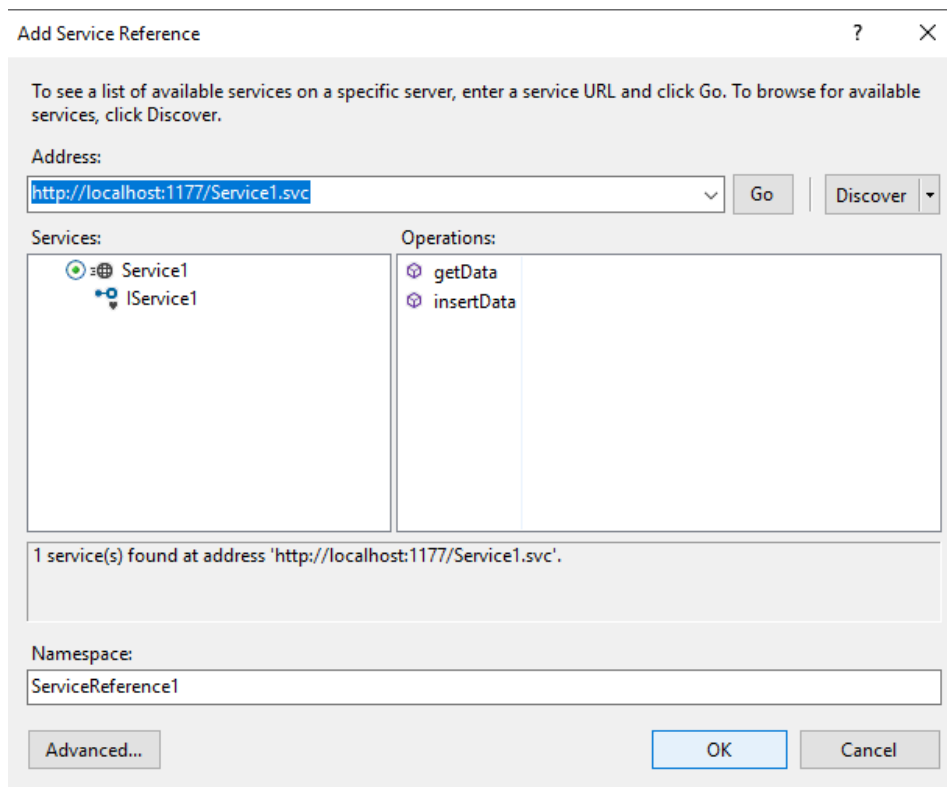


Copy the Link(<http://localhost:1177/Service1.svc>)



File -> New Web Site -> ASP.NET Empty Web Site (WebClient3)

Add -> Add Service Reference



Add-> New->web Form(WebClient4)

WebClient4.aspx.cs

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

public partial class WebClient4 : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        ServiceReference1.Service1Client sc = new ServiceReference1.Service1Client();
        DataSet ds = new DataSet();
        ds = sc.getData();
        GridView1.DataSource = ds;
        GridView1.DataBind();
    }
    protected void Insert_Click(object sender, EventArgs e)
    {
        ServiceReference1.Service1Client sc = new ServiceReference1.Service1Client();
        sc.insertData(Convert.ToInt32(t1.Text), t2.Text, Convert.ToInt32(t3.Text));
        l4.Text = "Data Inserted !!!!!";
    }
}
```

Output

Service1 Service x localhost:1268/WebClient4.aspx x +

localhost:1268/WebClient4.aspx

ID	Name	Salary
1	Iron Man	300000
2	Captian America	250000
3	Thor	200000
4	Hulk	150000
5	Black Widow	100000

Service1 Service x localhost:1268/WebClient4.aspx x +

localhost:1268/WebClient4.aspx

ID	Name	Salary
1	Iron Man	300000
2	Captian America	250000
3	Thor	200000
4	Hulk	150000
5	Black Widow	100000

Id

Name

Salary

Data Inserted !!!!!

Service1 Service x localhost:1268/WebClient4.aspx x +

localhost:1268/WebClient4.aspx

ID	Name	Salary
1	Iron Man	300000
2	Captian America	250000
3	Thor	200000
4	Hulk	150000
5	Black Widow	100000
6	Ant-Man	50000

Id

Name

Salary