ASE LAB 4 REPORT

In the Lab 4 I have implemented the Restful web Service and WSDL web service and also created a web client service application using the SOAP based web service and successfully deployed into the cloud.

**Restful Web service:**

1. In the first step I have created a project for Simple Arithmetic operations like

1. Addition

2. Subtraction

3. Multiplication

4. Division

1. Created the project in WCF Service Application.
2. Later, the code for arithmetic operations is written in the lservice.cs file.
3. Created the GET, POST service in the service1.cs file.
4. By right clicking on the Service1.svc file “view in markup” is selected and the code is changed.
5. Later on by right clicking on the Service1.svc file “view in browser” the code is executed in the local browser and required is output is displayed.

**Deploying it into the cloud:**

1. First the complete file of the project is copied from the local system into the remote desktop.
2. Then the file is once again copied in to the c->inetpub->wwwroot->Asp\_client
3. The IIS manager is opened and the file is converted in to the application, then the permissions to the application is changed
4. The service1.svc file is executed by right clicking and selecting the browse
5. The required output is shown even in the remote desktop.

**WSDL service:**

1. In the first step a new website is created then a new item “web service” is added.
2. The code is written for the webservice.cs file and the file is executed.
3. The link of the respective opened is copied and added as the web reference to the website
4. A webform is created to the simple arithmetic operations.
5. The necessary changes are made to the web.config file.
6. The website is executed and the desired output is achieved.

**Deploying it into the cloud:**

1. The complete website file is copied into the remote desktop.
2. Then, the file is copied to the c->inetpub->wwwroot->Asp\_client
3. IIS Manager is opened and the file is converted in to the application.
4. The access permissions are changed.
5. Then the project is executed and the desired output is achieved.

**Web client Application:**

1. A new web site is created and a new item is a selected and a web service is added.
2. The required default.aspx file is opened and required web form is designed.
3. Then c# code is written in order to access the web form in the default.aspx.cs file
4. A web reference of the SOAP based web reference is added.
5. The resultant change can be seen in the web.config file.
6. Then the web site is build and it is started without debugging.
7. Then the desired output is obtained.

**Deploying it into the cloud:**

1. The complete web client application is file is copied into the remote desktop.
2. Then, the file is copied to the c->inetpub->wwwroot->Asp\_client
3. IIS Manager is opened and the file is converted in to the application.
4. The access permissions are changed.
5. Then the project is executed and the desired output is achieved.

This is the web reference link which is used in the web client application:

<http://170.224.169.112/aspnet_client/WebSite11/WebService.asmx>

**JSFIDDLE:**

The web client application is done in jsfiddle including the html and JavaScript code.

**HTML code:**

<h1>CALCULATOR</h1>

<input type="text" id="number1" placeholder="Enter the First Number"></input>

<input type="text" id="number2" placeholder="Enter the Second Number"></input>

<br>

<input type="button" class="add" value="Addition"></input>

<input type="button" class="subtract" value="Subtraction"></input>

<input type="button" class="mul" value="Multiplication"></input>

<input type="button" class="div" value="Division"></input>

<br>

<p> Result <p>

<input type="text" id="Result"></input>

**JAVASCRIPT:**

var echo = function(dataPass) {

$.ajax({

type: "POST",

url: "/echo/json/",

data: dataPass,

cache: false,

success: function(json) {

alert("Result=" + json.Id);

}

});

};

$('.add').live('click', function() {

$.get("http://170.224.169.112/aspnet\_client/WebSite11/WebService.asmx", function(data) {

var json = {

json: JSON.stringify(data),

delay: 1

};

echo(json);

});

});

$('.subtract').live('click', function() {

$.get("http://170.224.169.112/aspnet\_client/WebSite11/WebService.asmx", function(data) {

var json = {

json: JSON.stringify(data),

delay: 1

};

echo(json);

});

});

$('.mul').live('click', function() {

$.get("http://170.224.169.112/aspnet\_client/WebSite11/WebService.asmx", function(data) {

var json = {

json: JSON.stringify(data),

delay: 1

};

echo(json);

});

});

$('.div').live('click', function() {

$.get("http://170.224.169.112/aspnet\_client/WebSite11/WebService.asmx", function(data) {

var json = {

json: JSON.stringify(data),

delay: 1

};

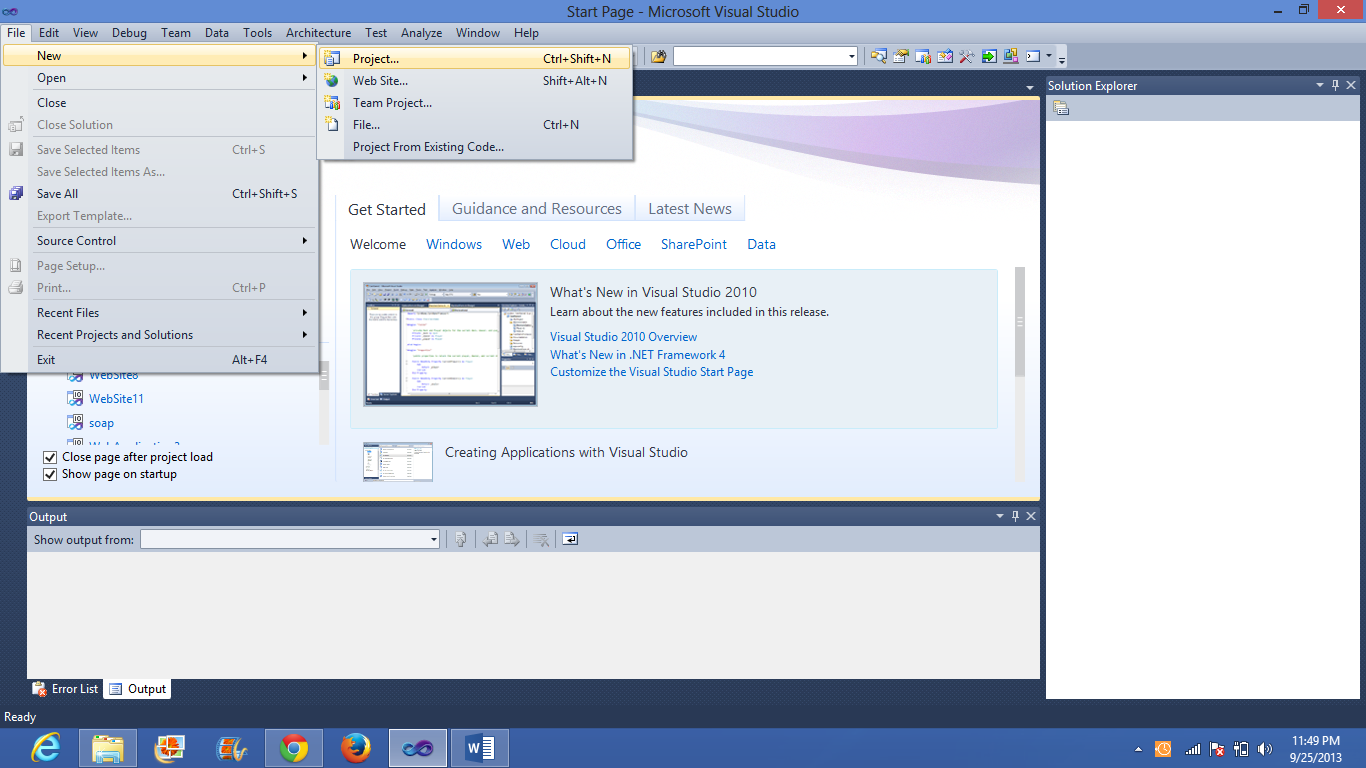
echo(json);

});

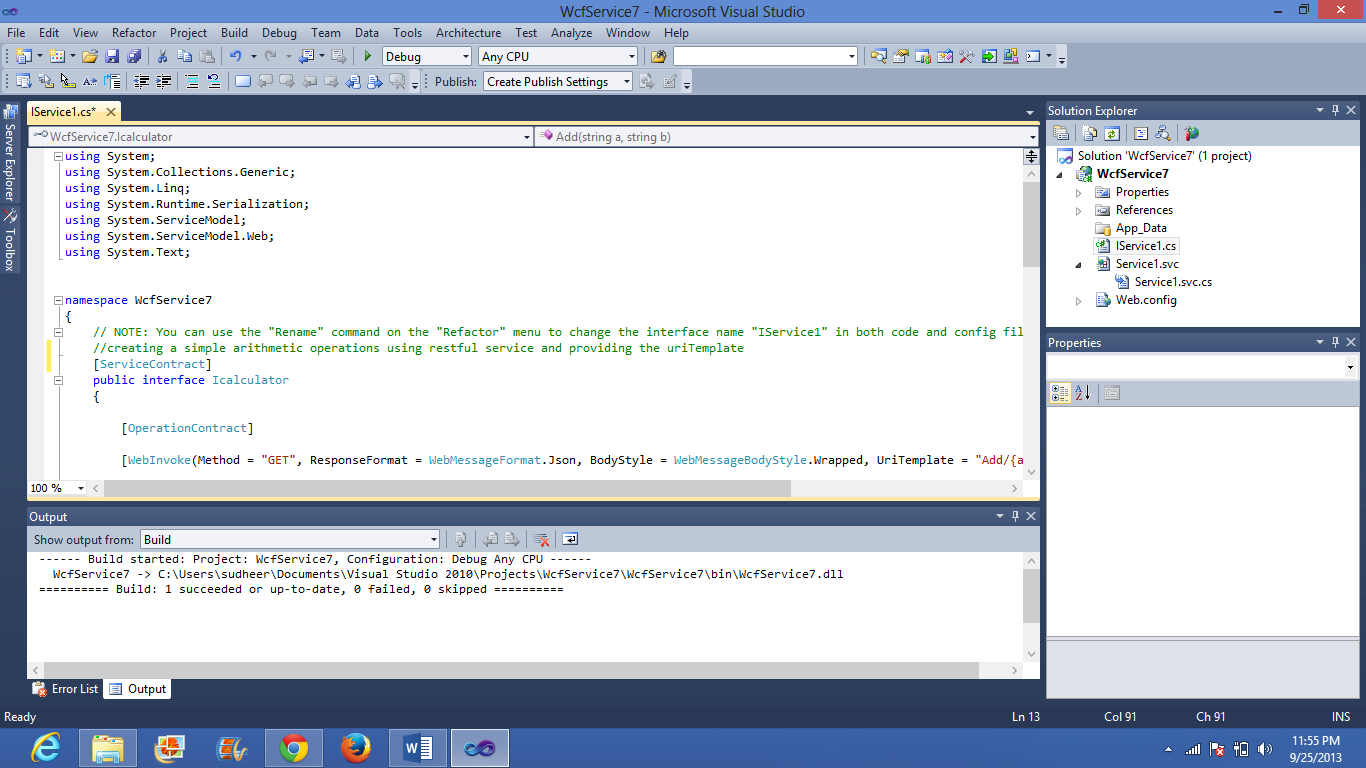
});

**Restful Web Service:**

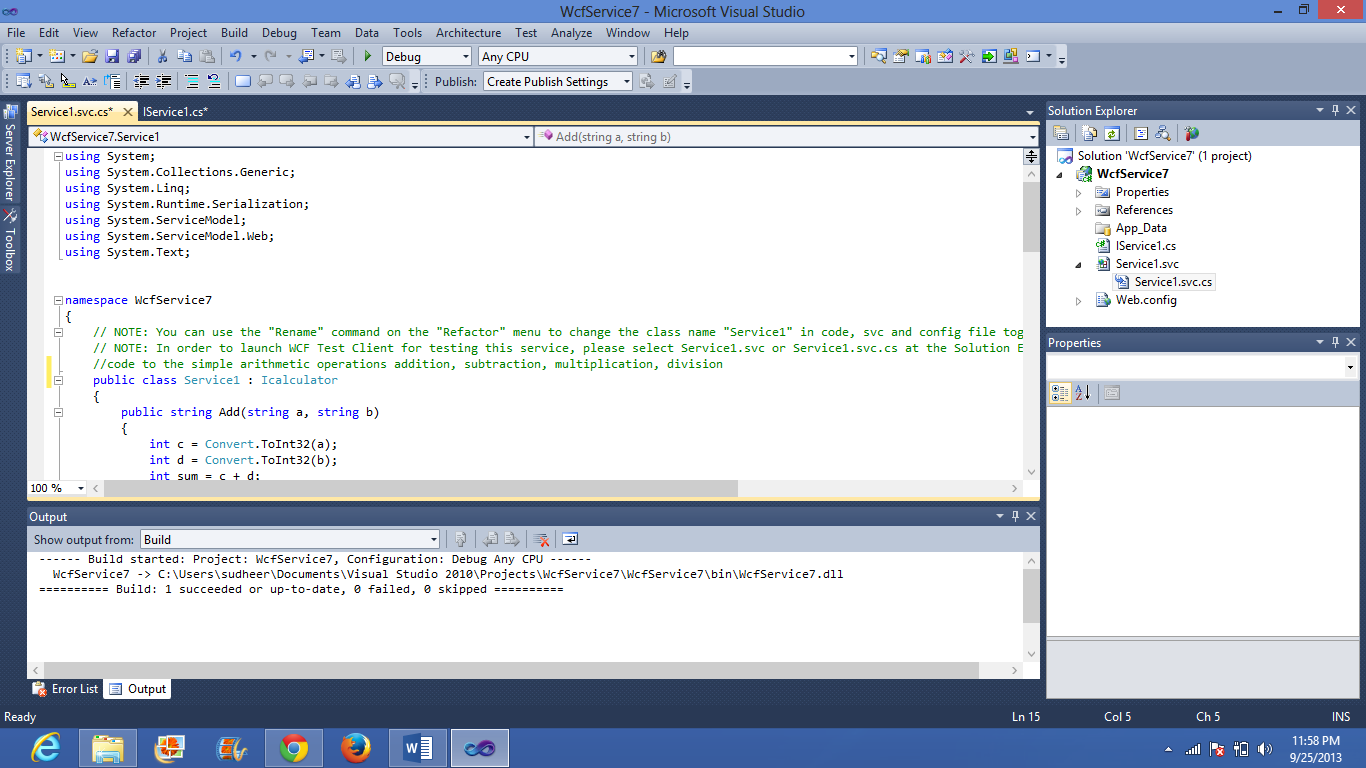
Creating a new project



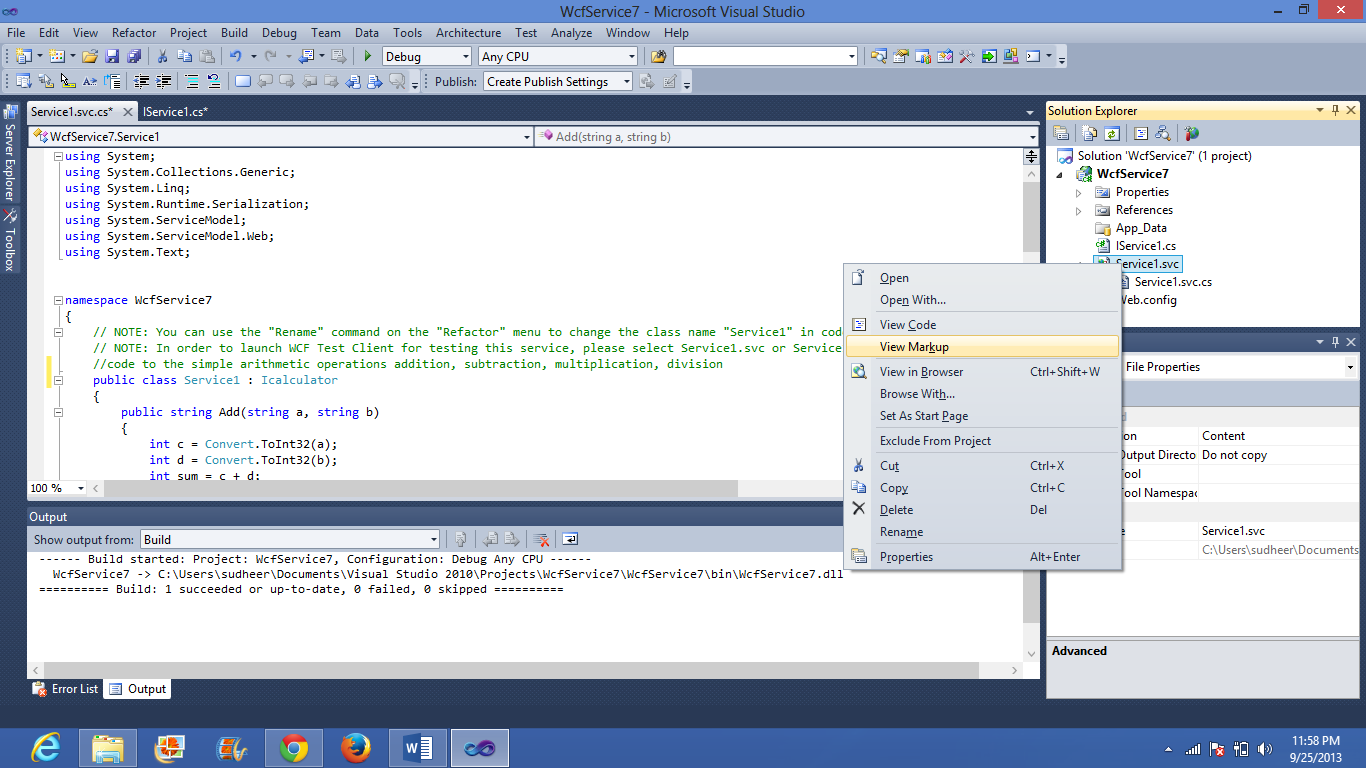
Code for lservice1.cs for simple arithmetic calculations



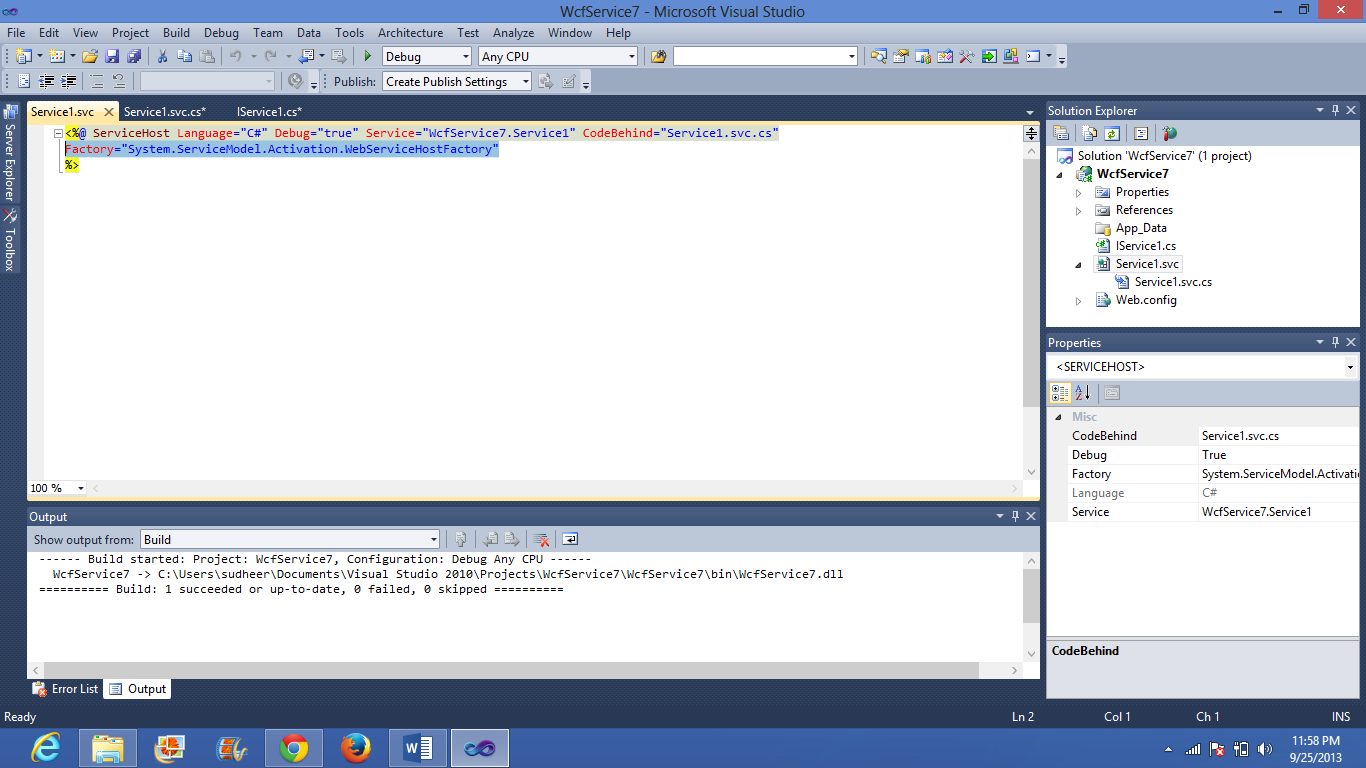
Service1.svc.cs code for calculator



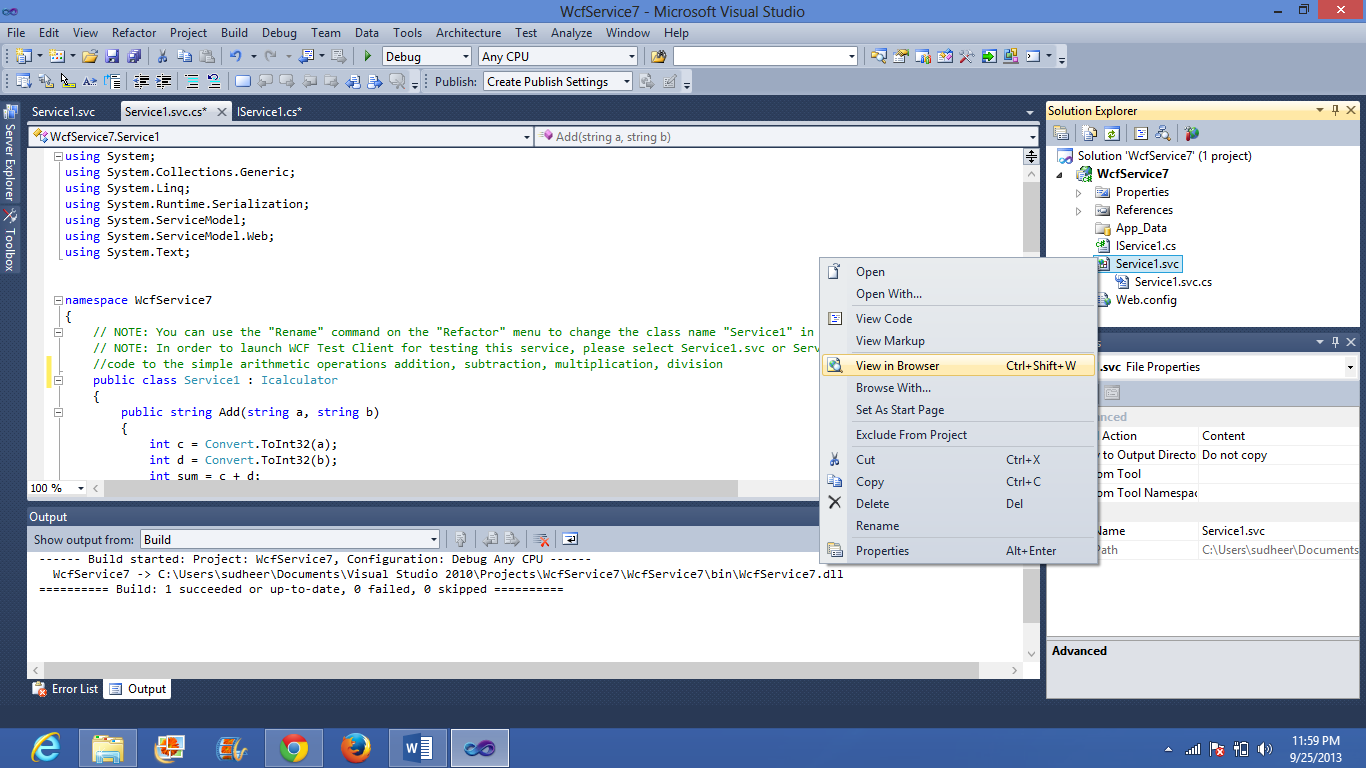
Right clicking on service1.svc and selecting the view in mark up



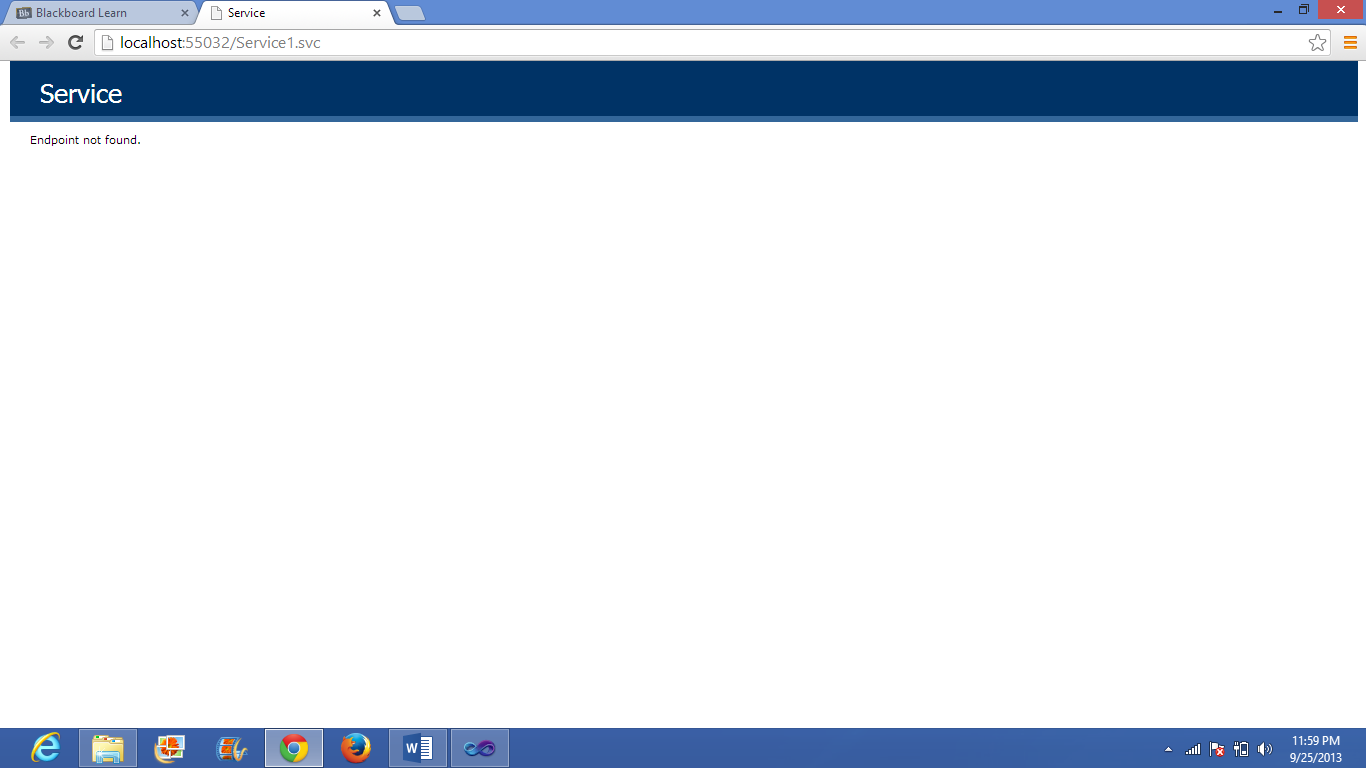
Adding this code



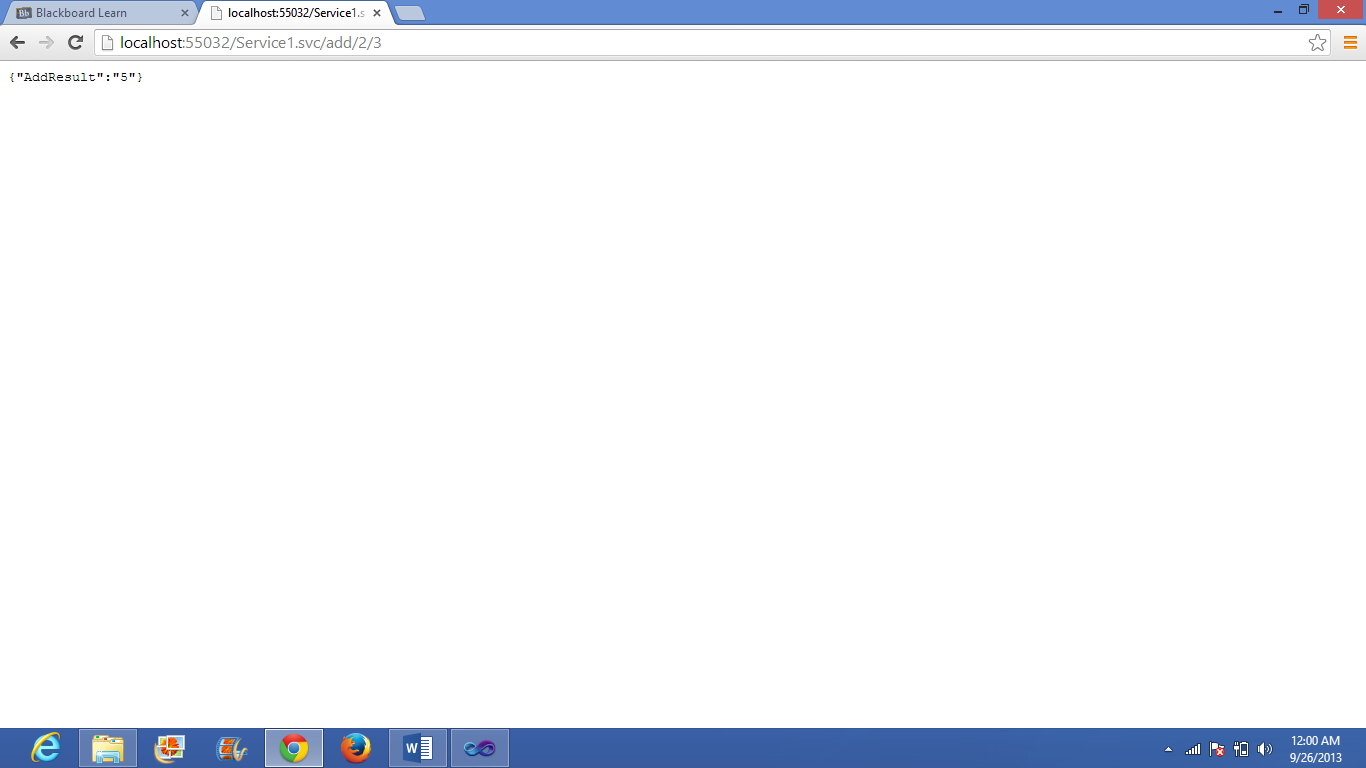
Right click on service1.svc and select view in browser



The following output displays in a web page



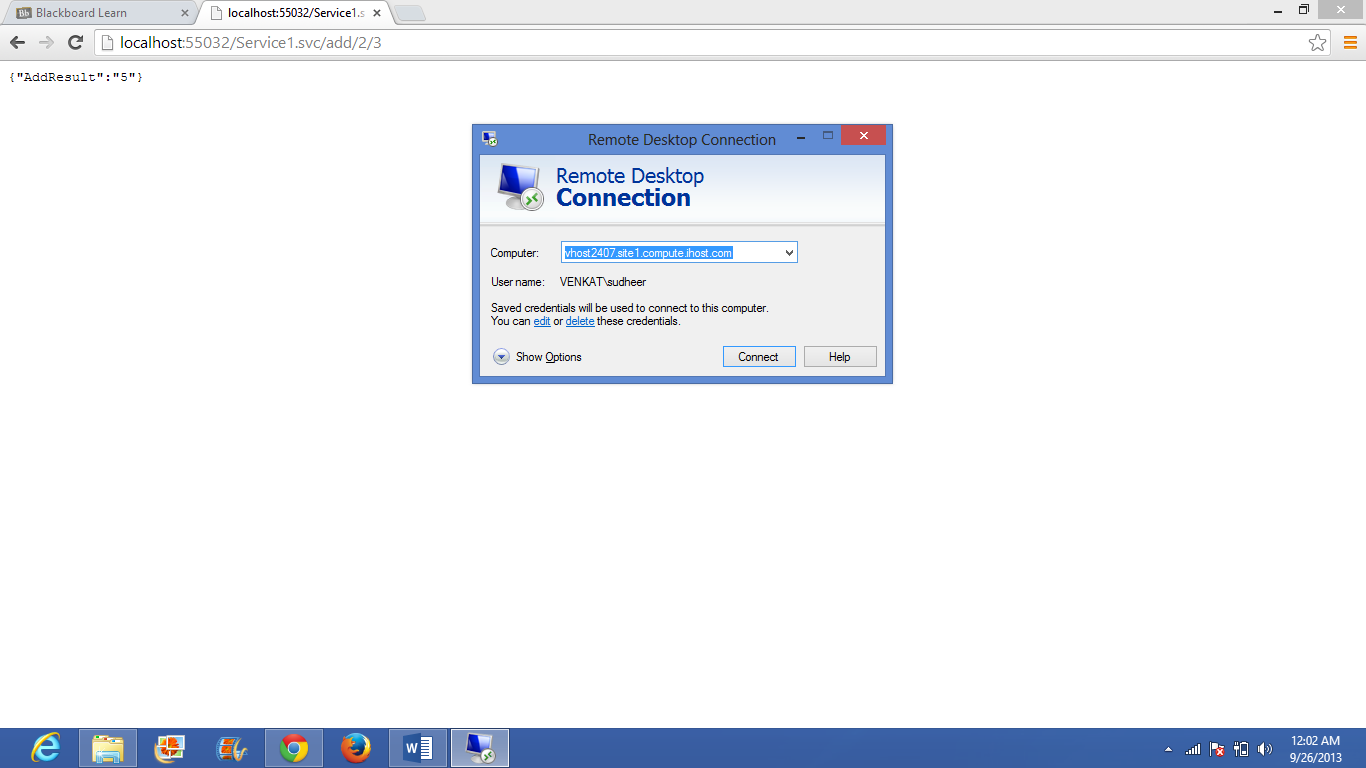
Adding “/add/2/3” to get the desired output



This is how we use the restful web service.

**Deploying the restful web service in to the cloud:**

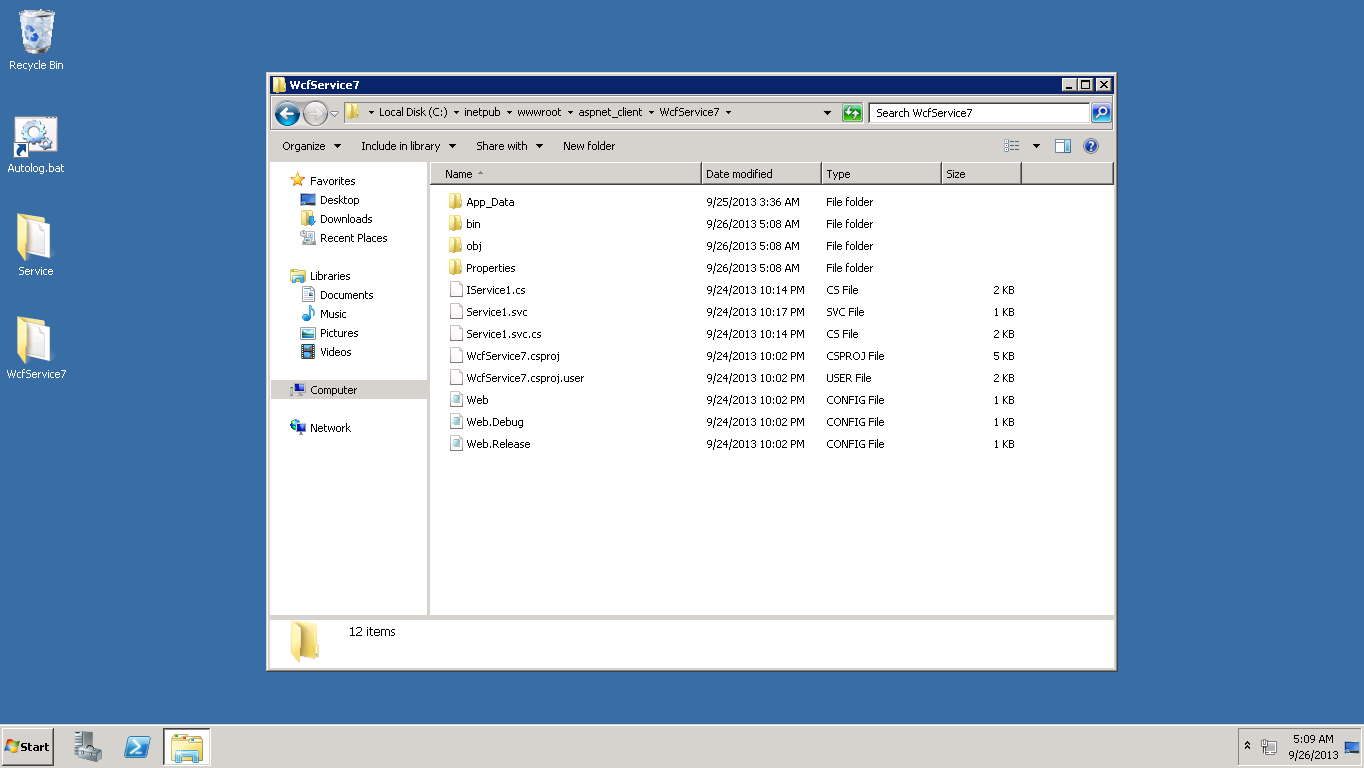
1. Open remote connection desktop and enter the instance id/IP address and credentials



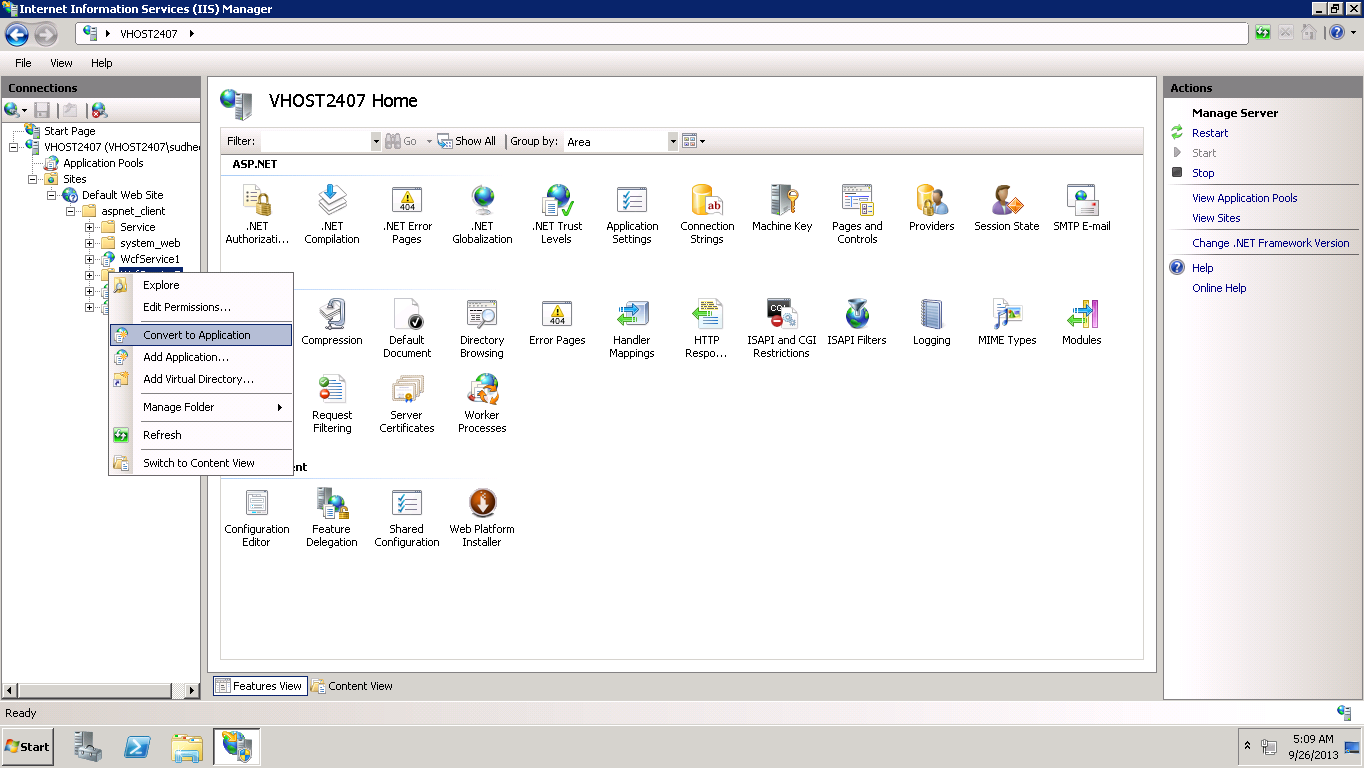
The following remote desktop appears and now the wcfservice file is copied into the remote desktop

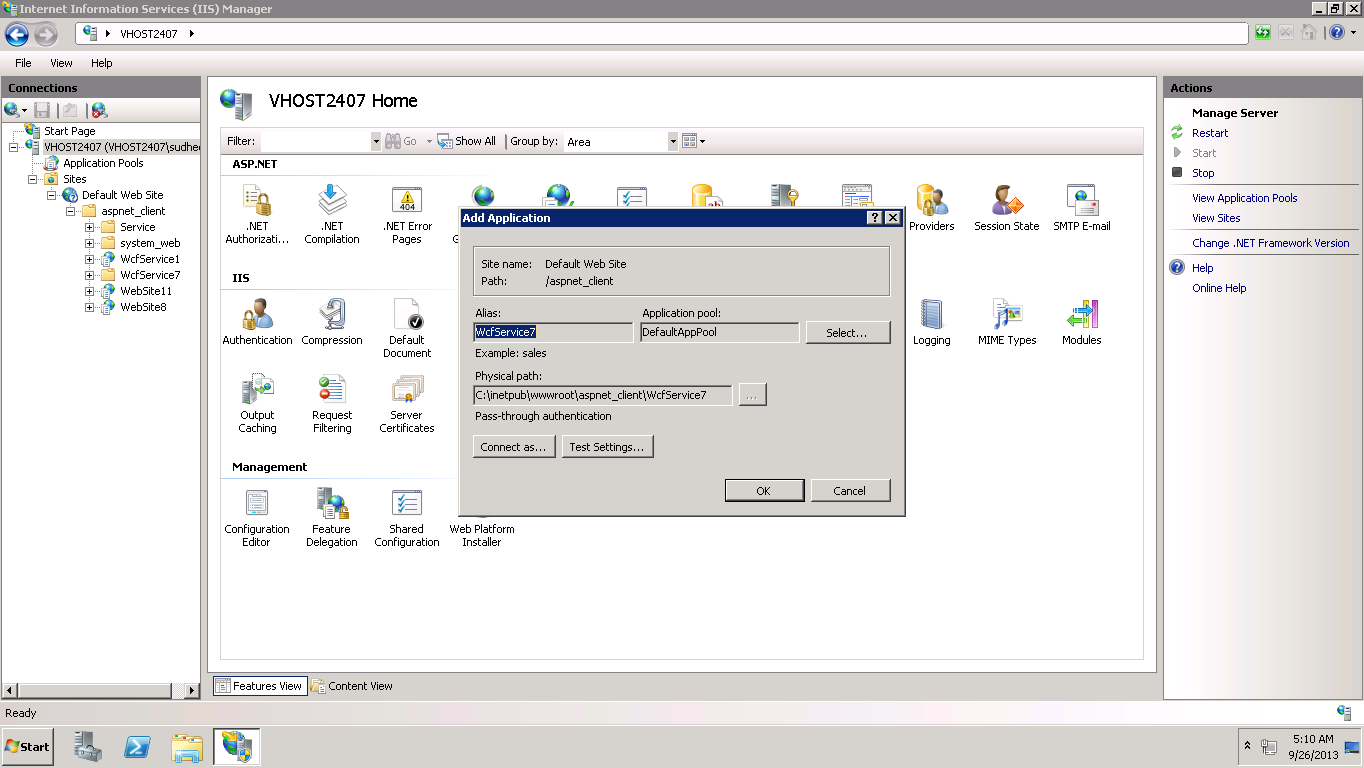


The file is copied into the c->inetpub->wwwroot->asp\_client

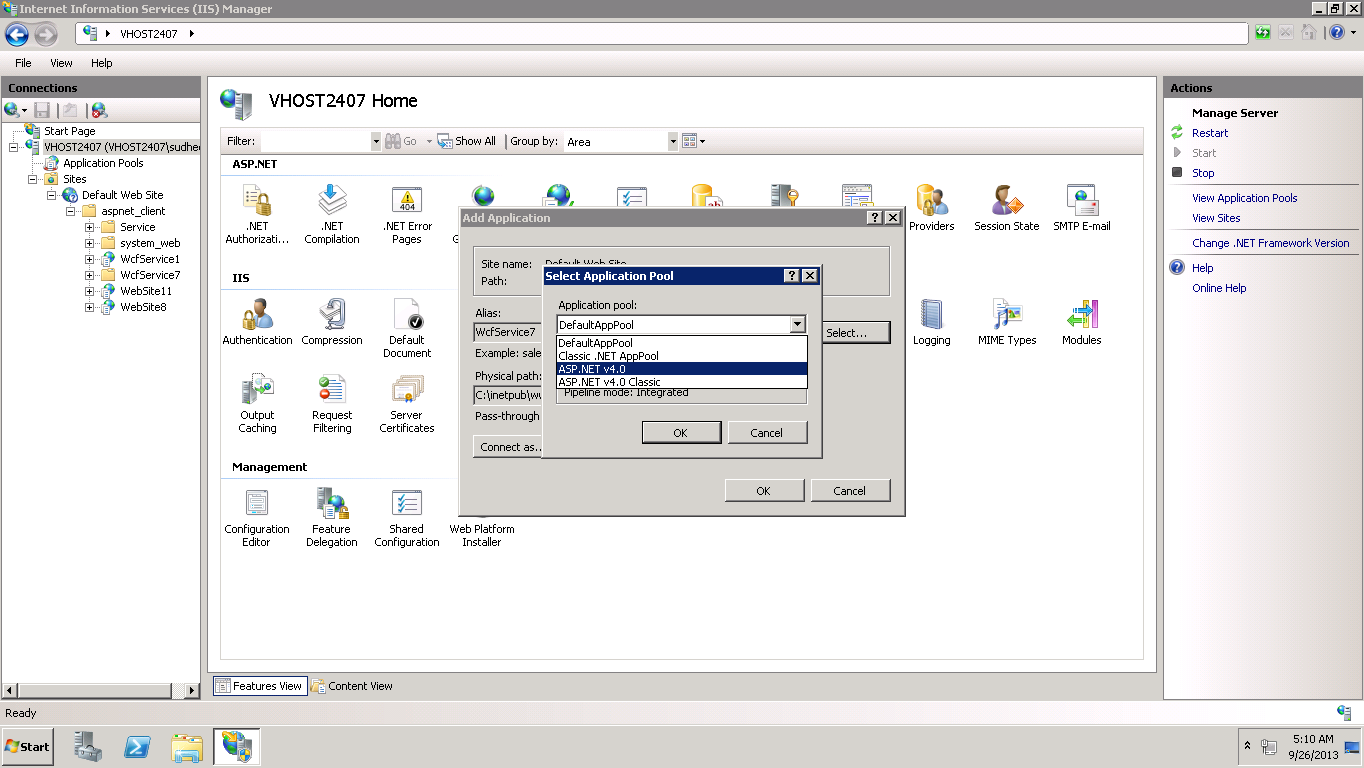


Then the IIS manager is opened and the wcfservice file converted into the application

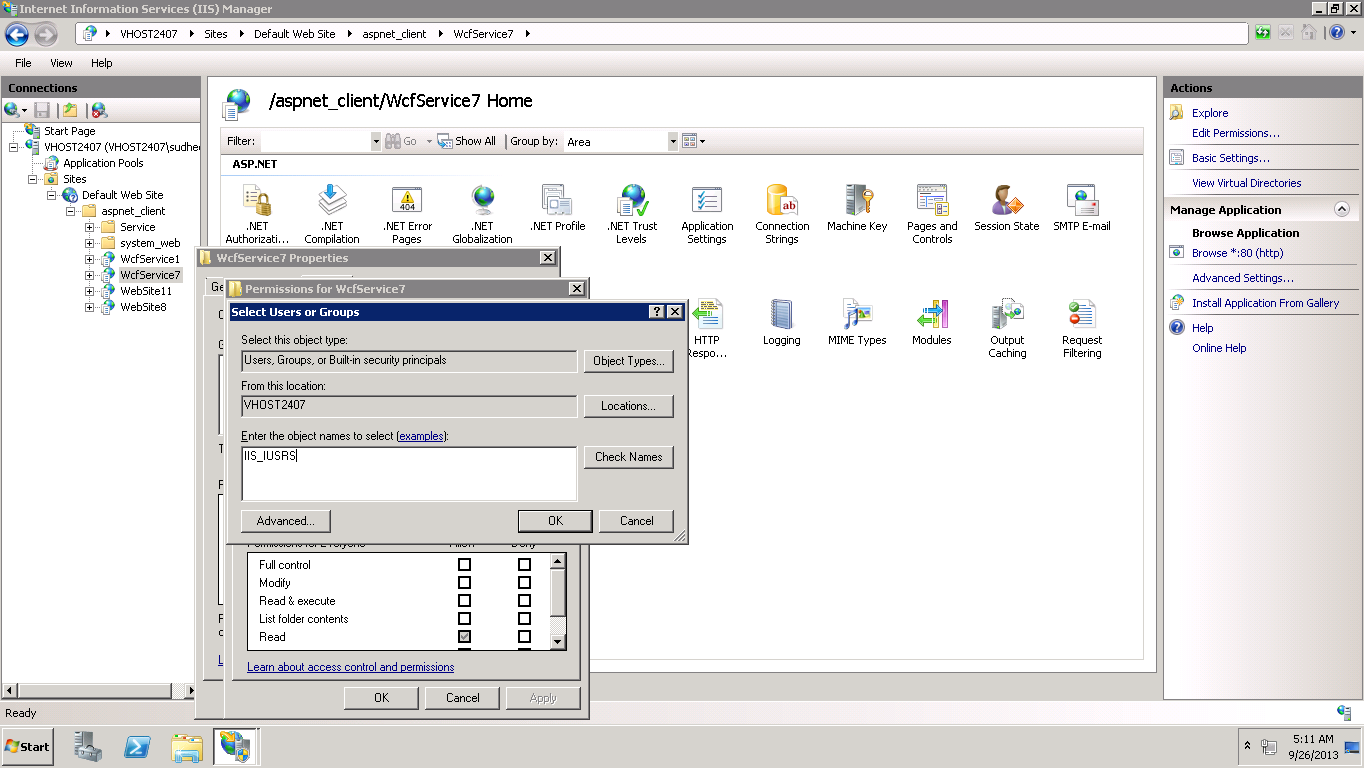


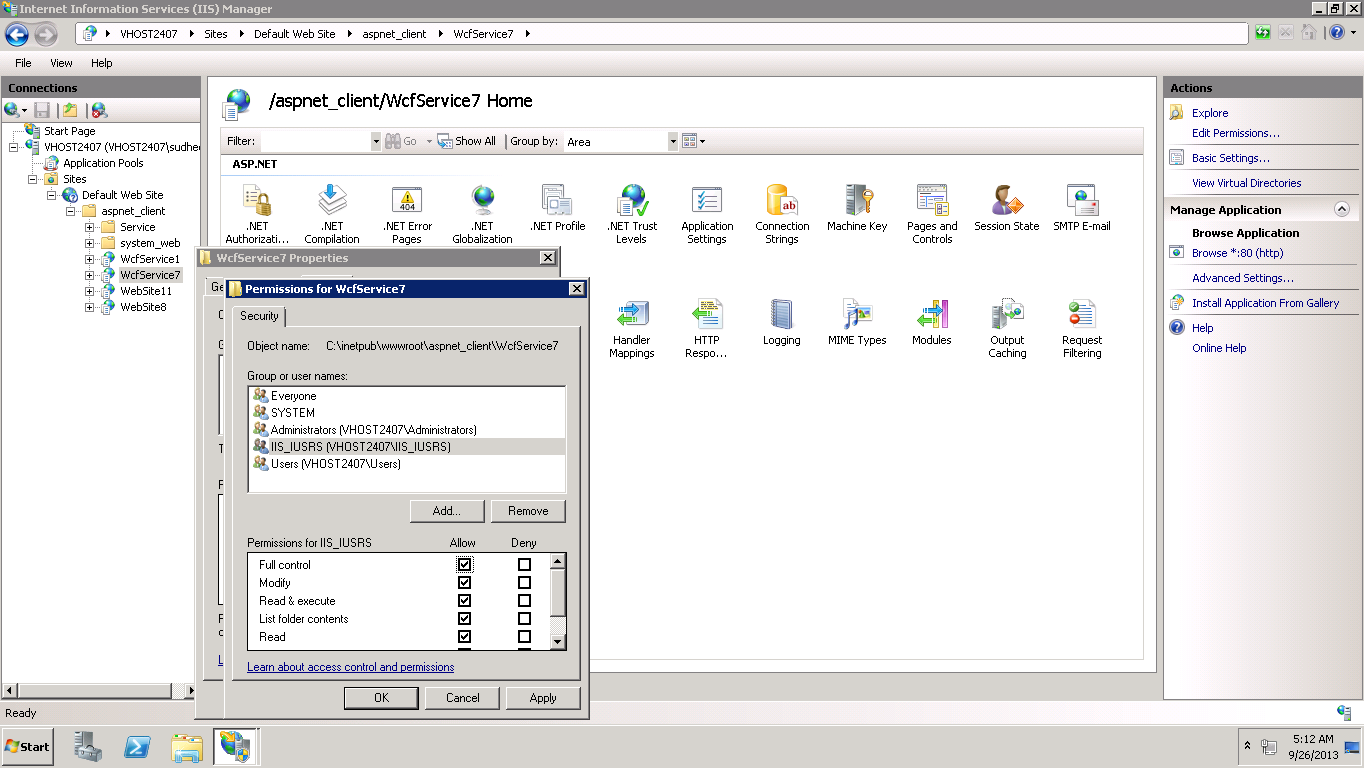


Even the application pool is selected to ASP.NET v4.0

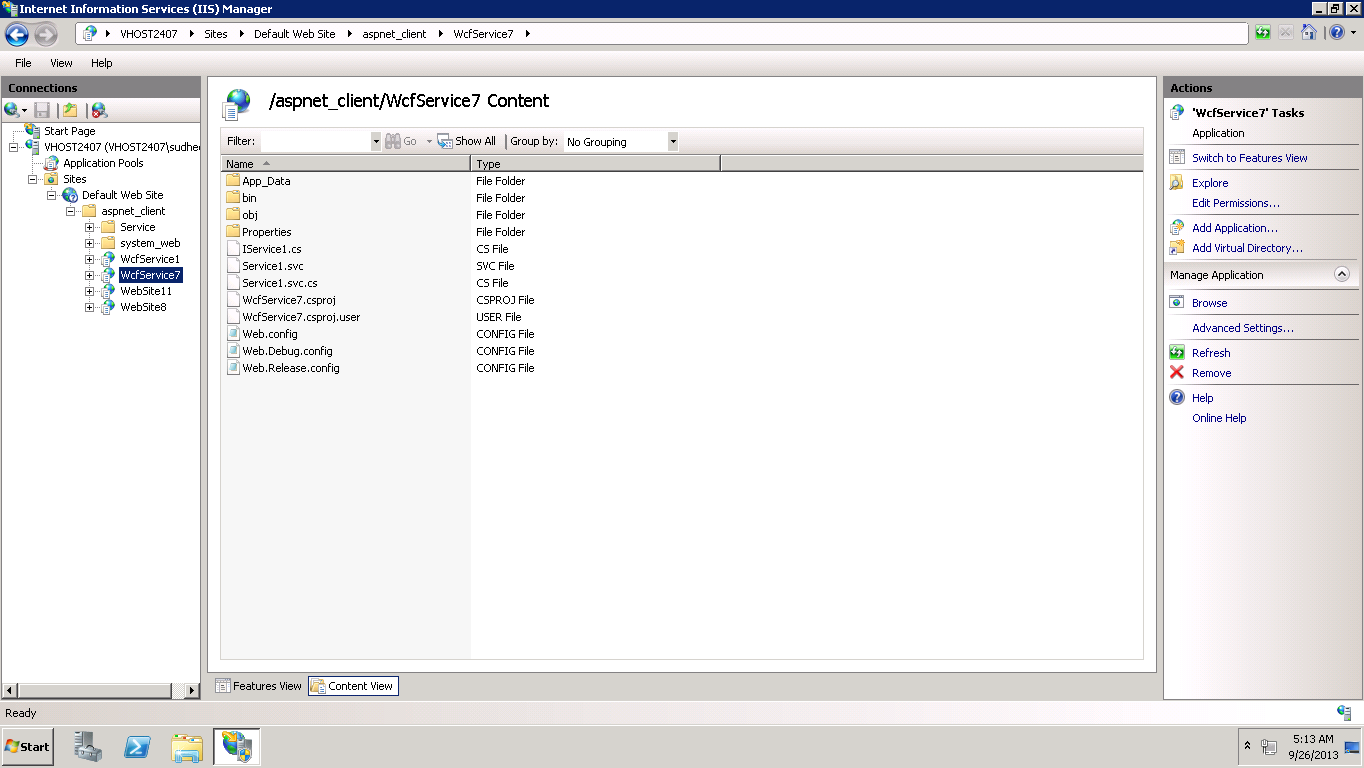


A new group is attached as “IIS\_IUSRS” and full control is accessed

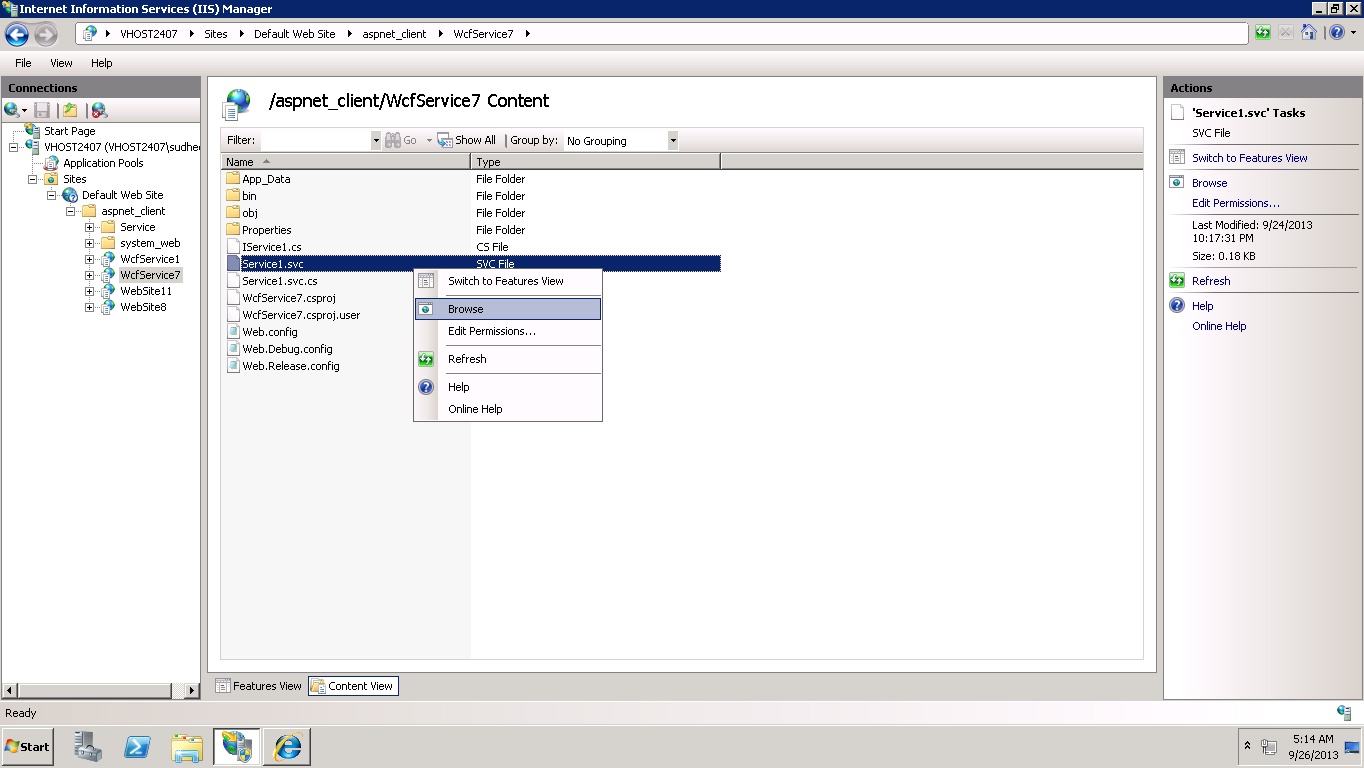




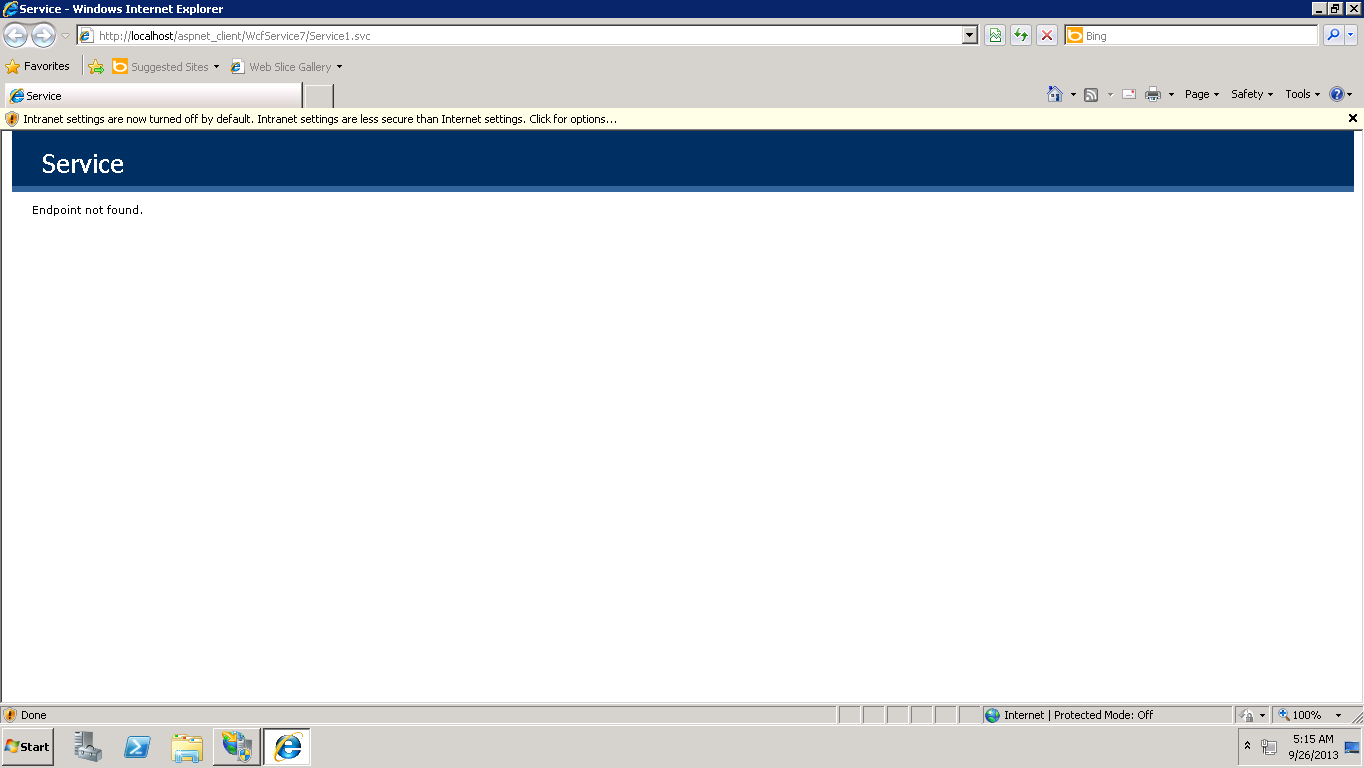
The content view of file is opened



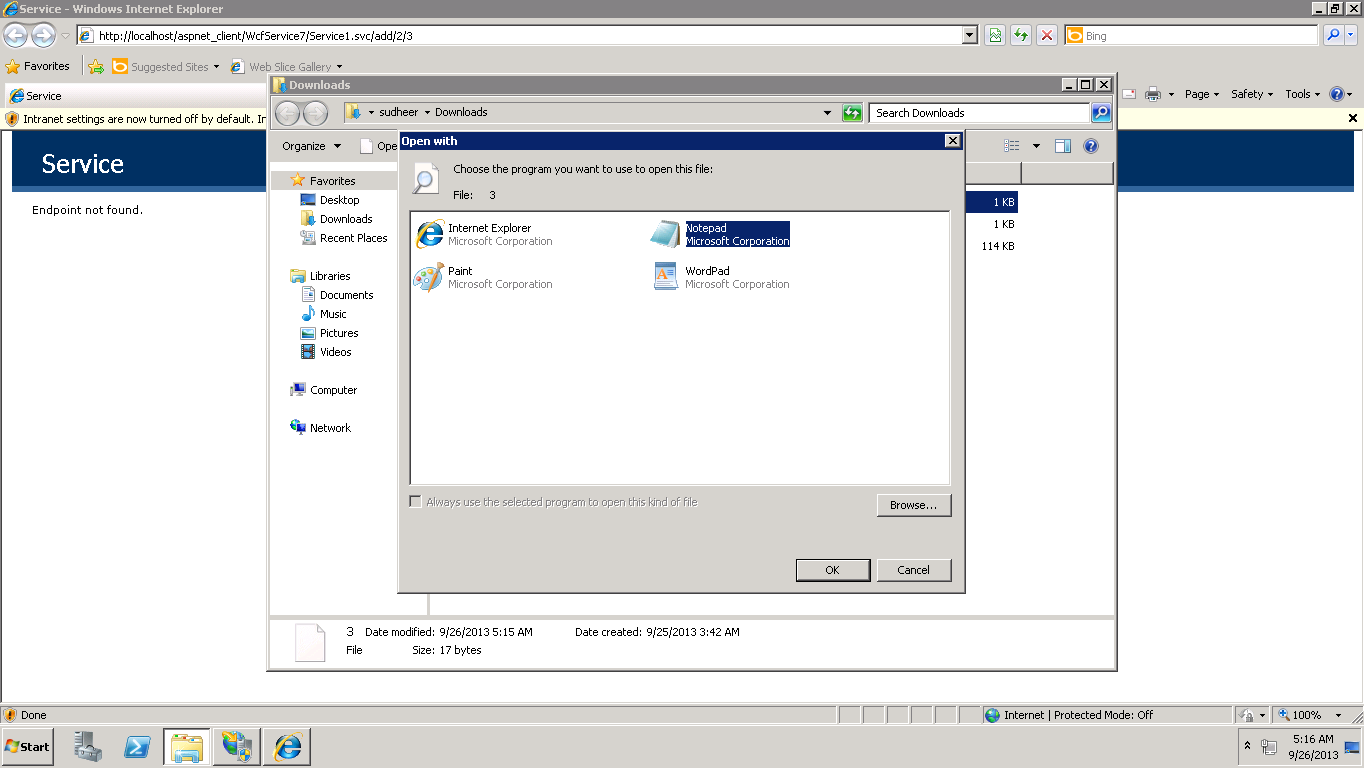
Right click on the service1.svc and browse is selected



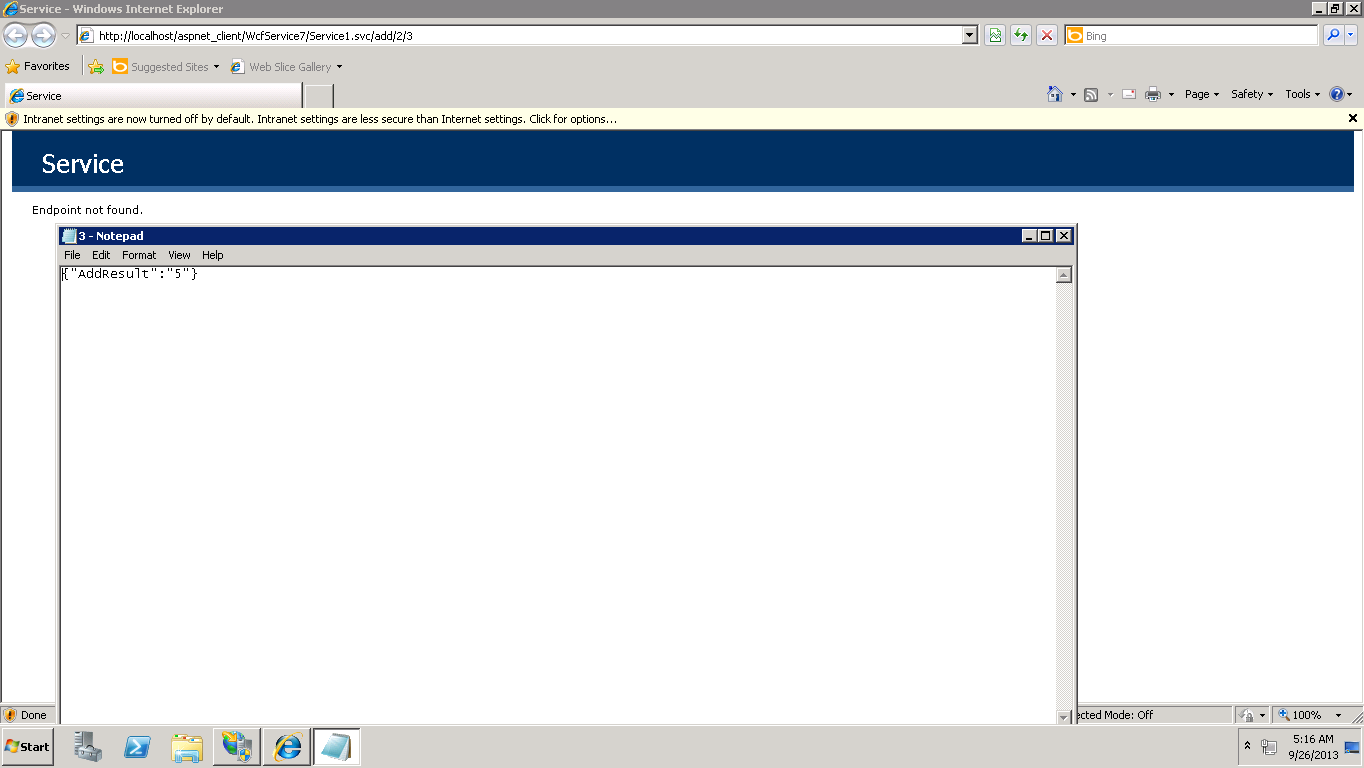
The following output appears



Then “add/2/3” is added and the file is opened in the notepad

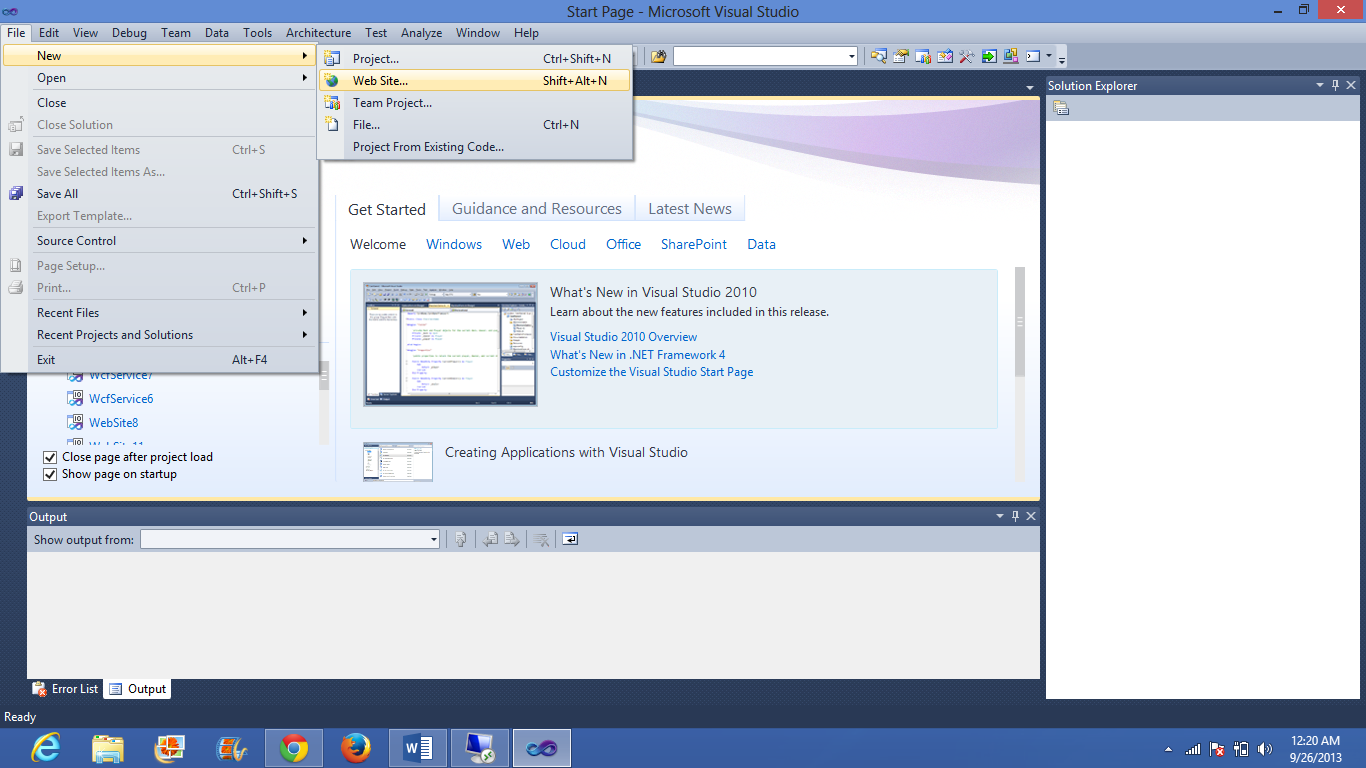


The desired output is achieved and the restful web service is successfully deployed into the cloud.

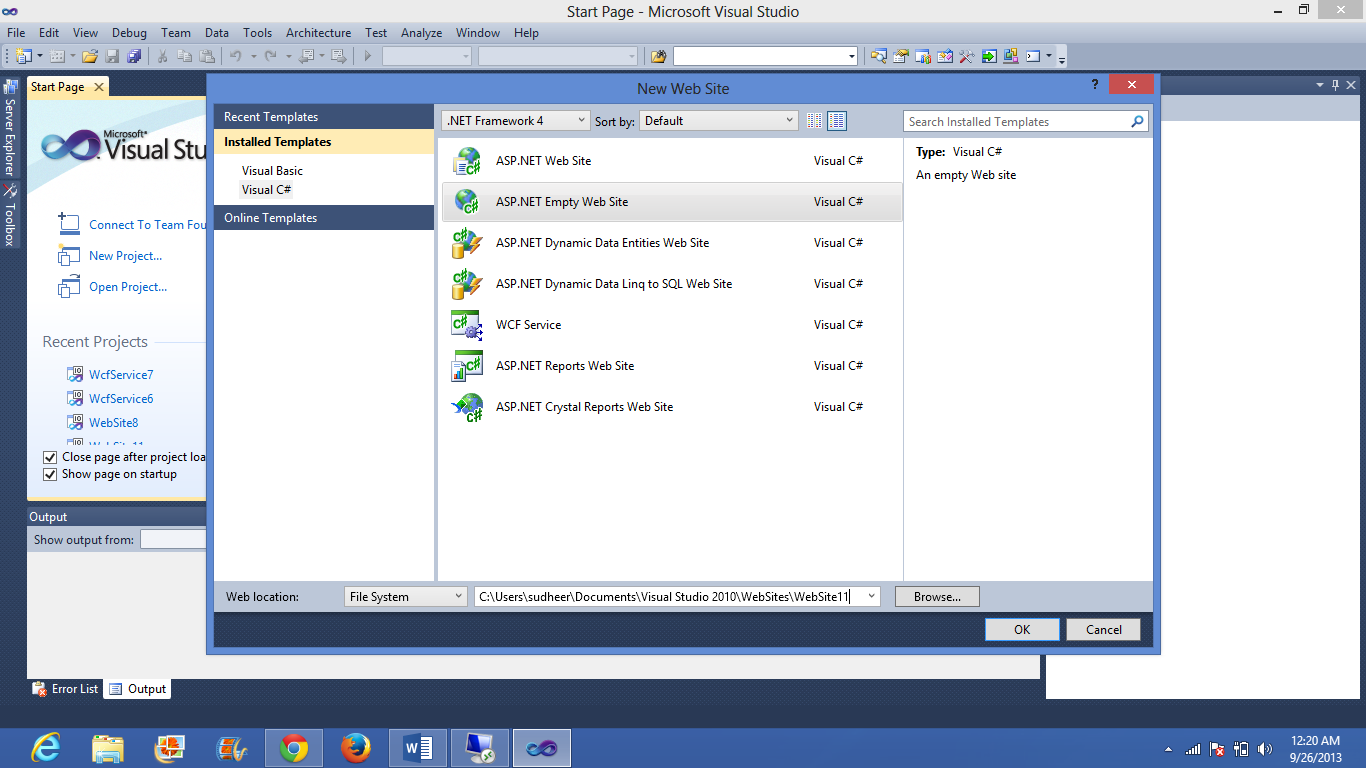


**SOAP based web service:**

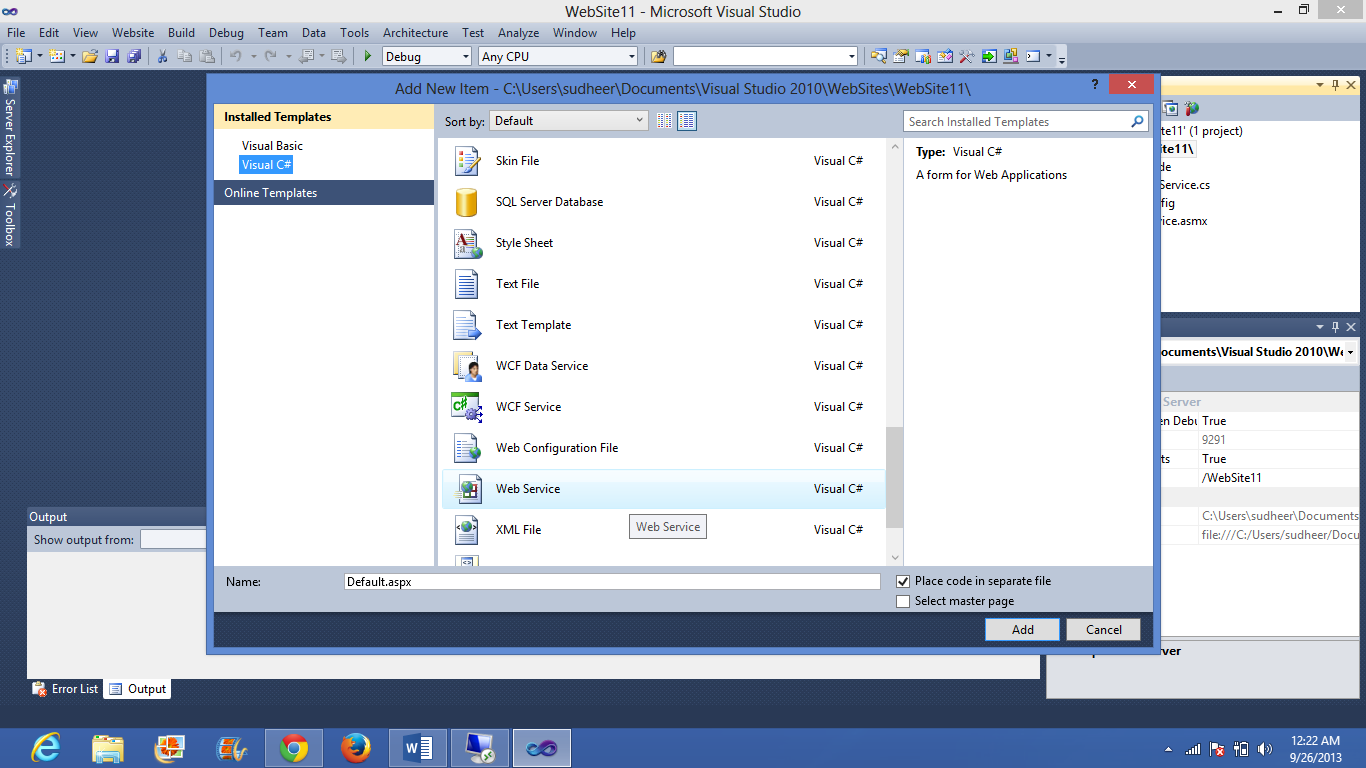
First a new website is created in the visual studio



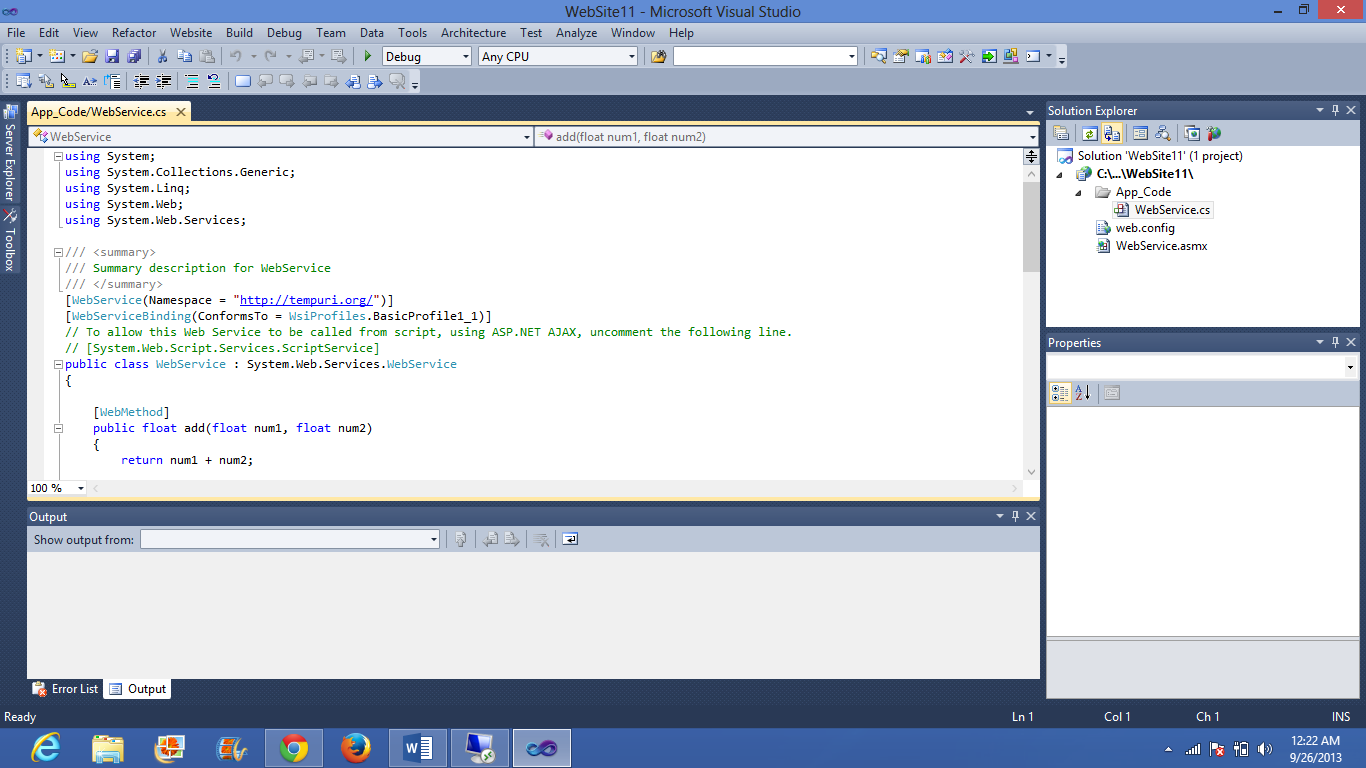
ASP.NET Empty web site is selected



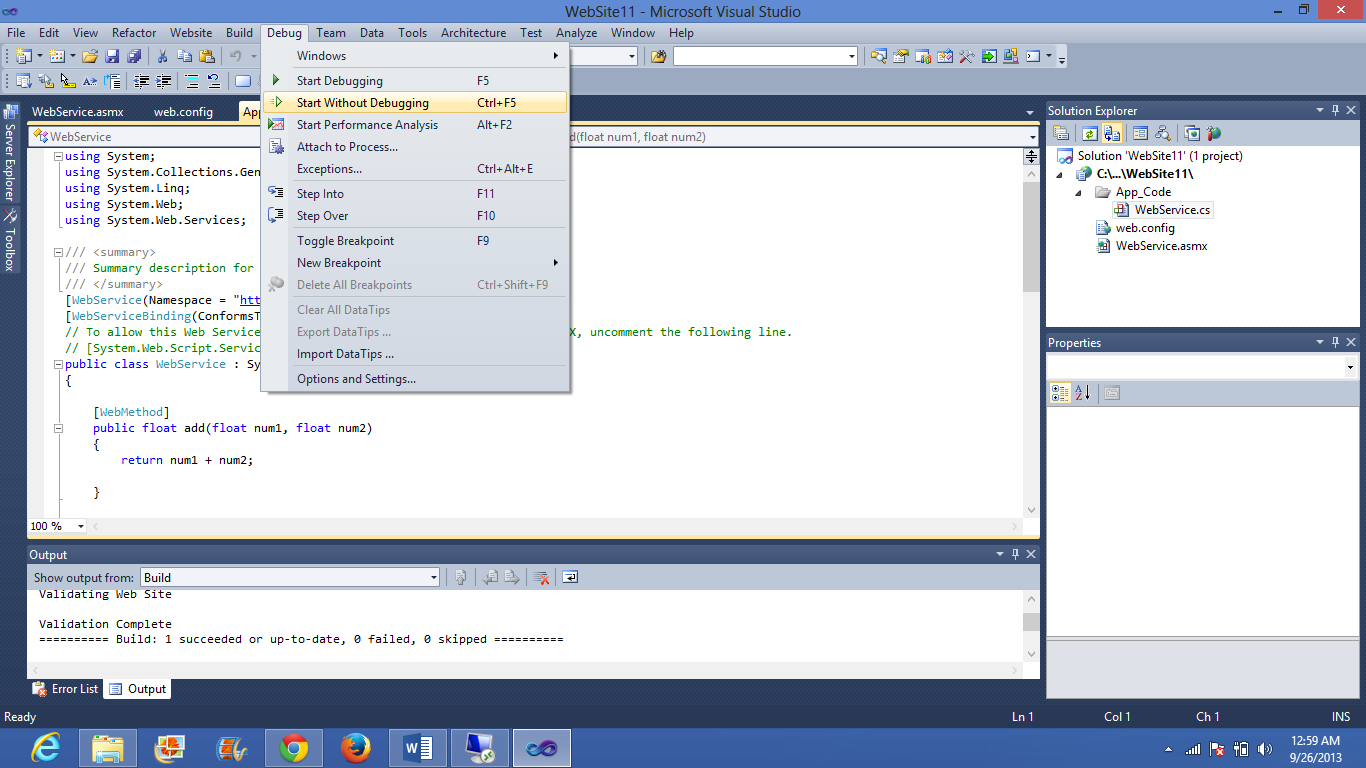
By right clicking on the website a add new item is selected and a web service is added



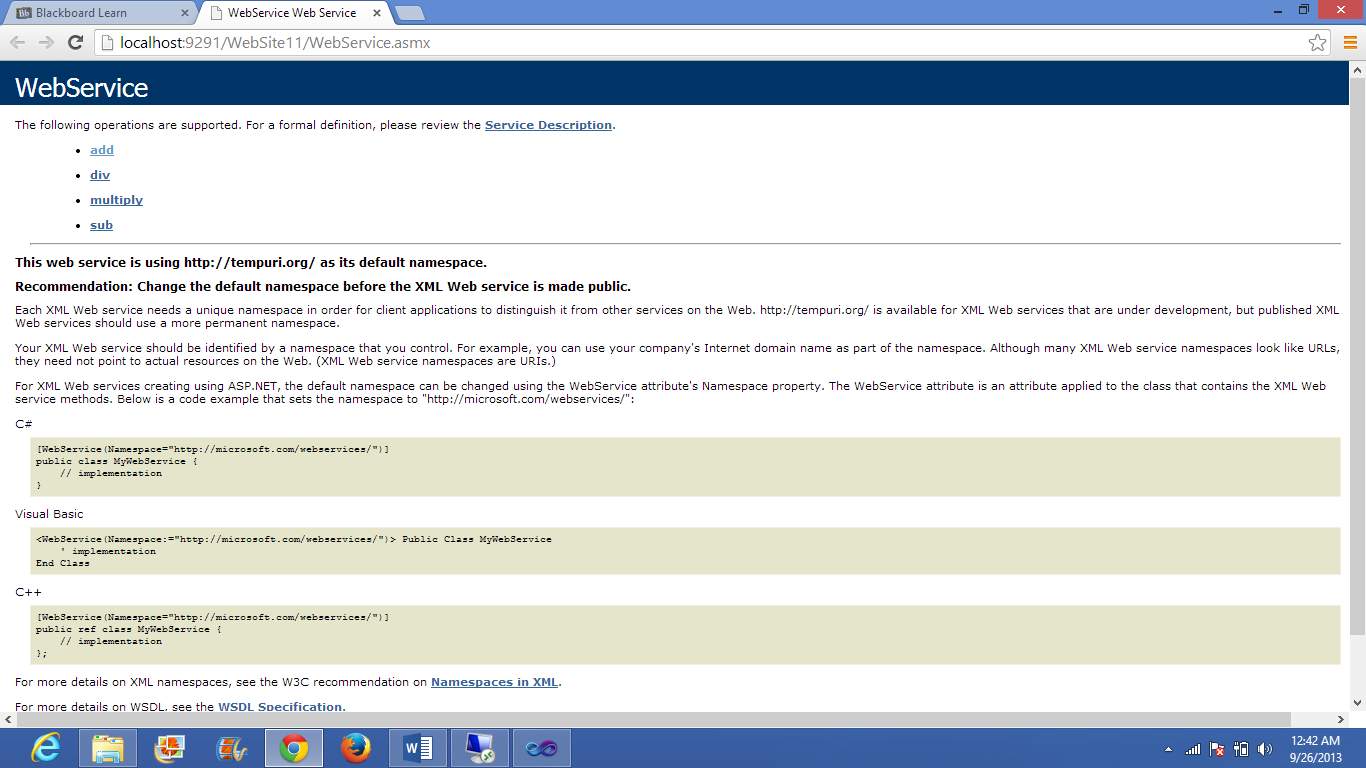
Webservice.cs is code is written to the arithmetic operations



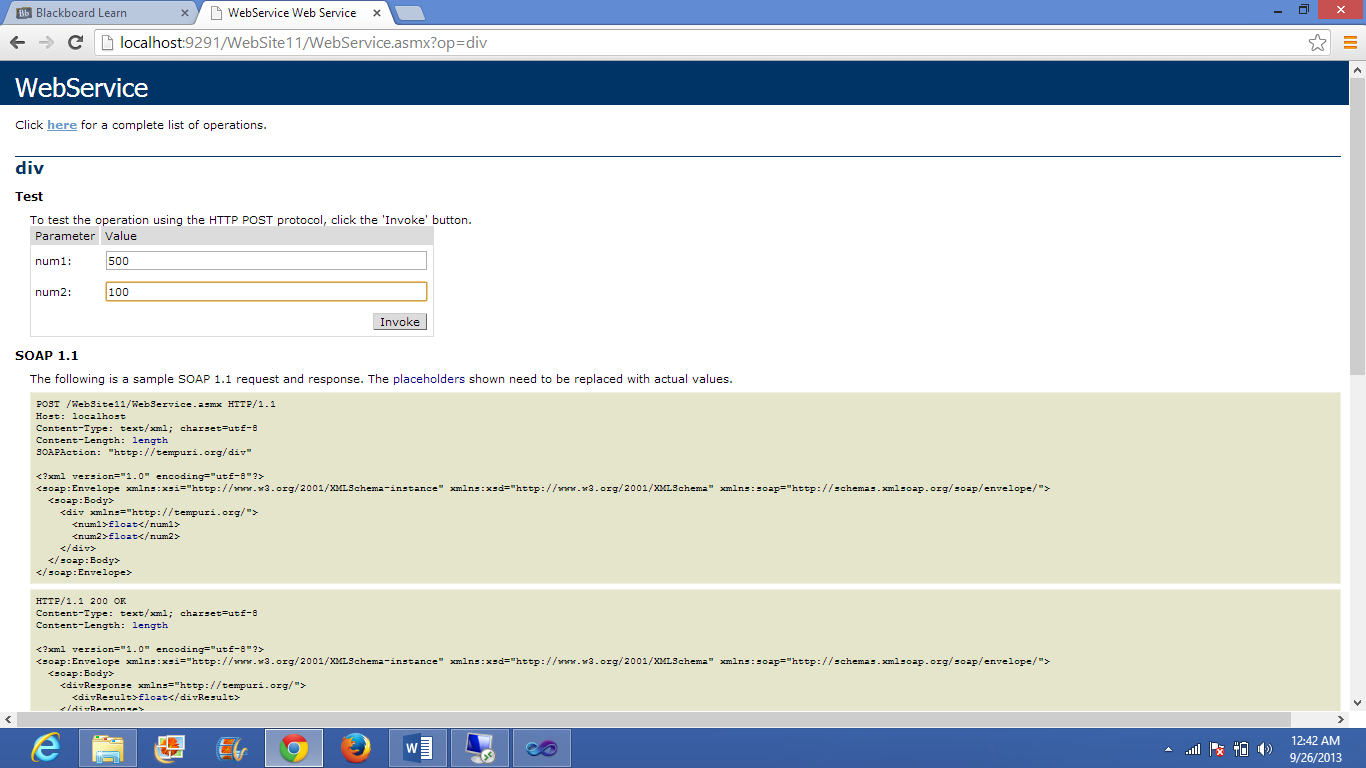
After clicking on the build and selecting the build solution, click on the debug and select start without debugging



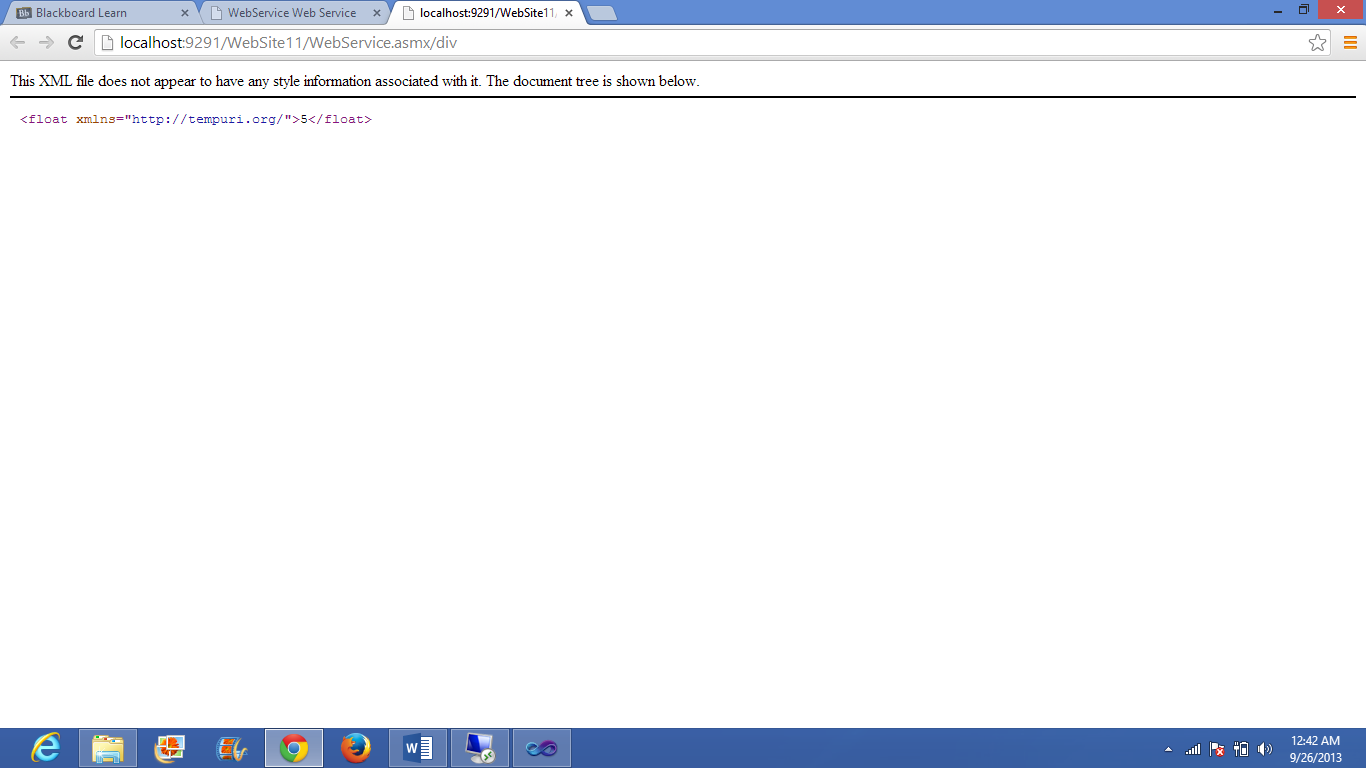
The following output is displayed in the web page



By selecting the division(div) operation and entering the numbers and clicking on the invoke button



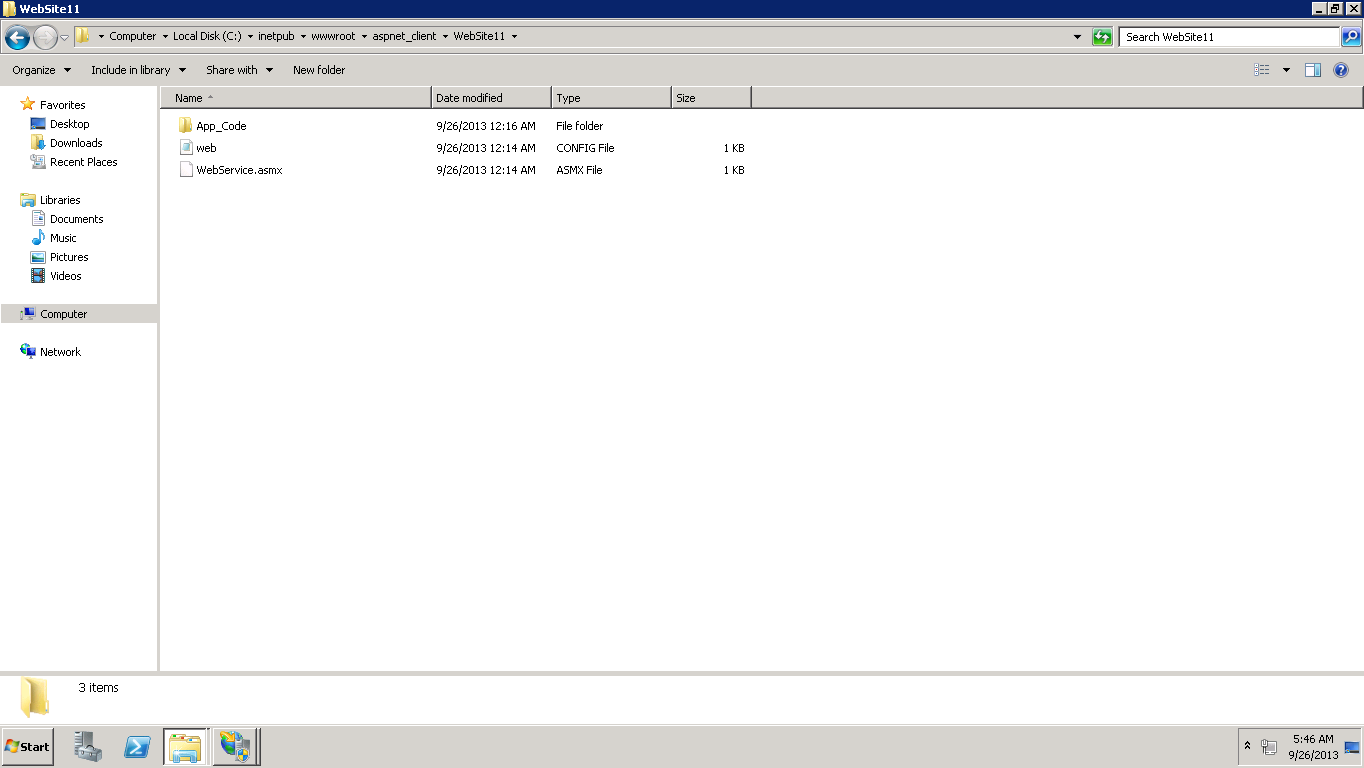
The following output is appeared



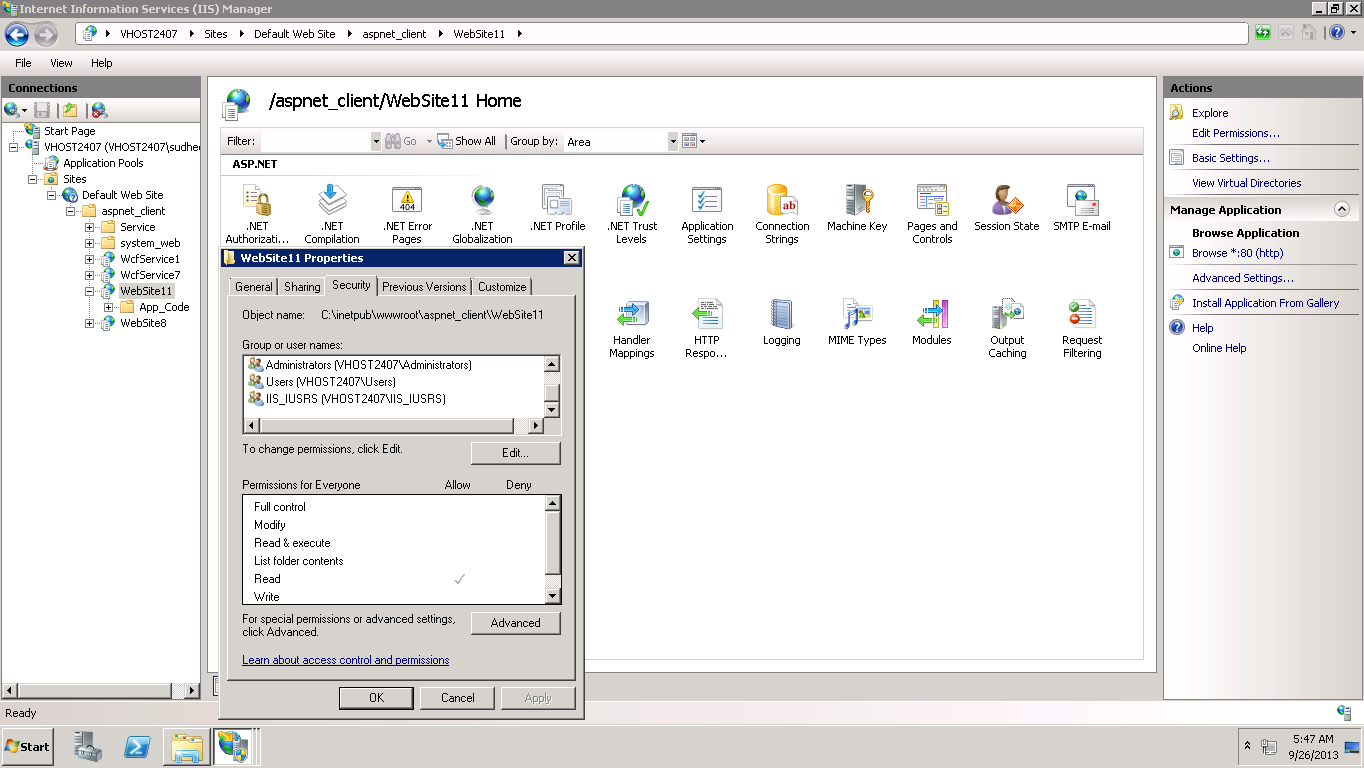
**Deploying the SOAP based web service in to the cloud:**

Copy the website file in to the remote desktop

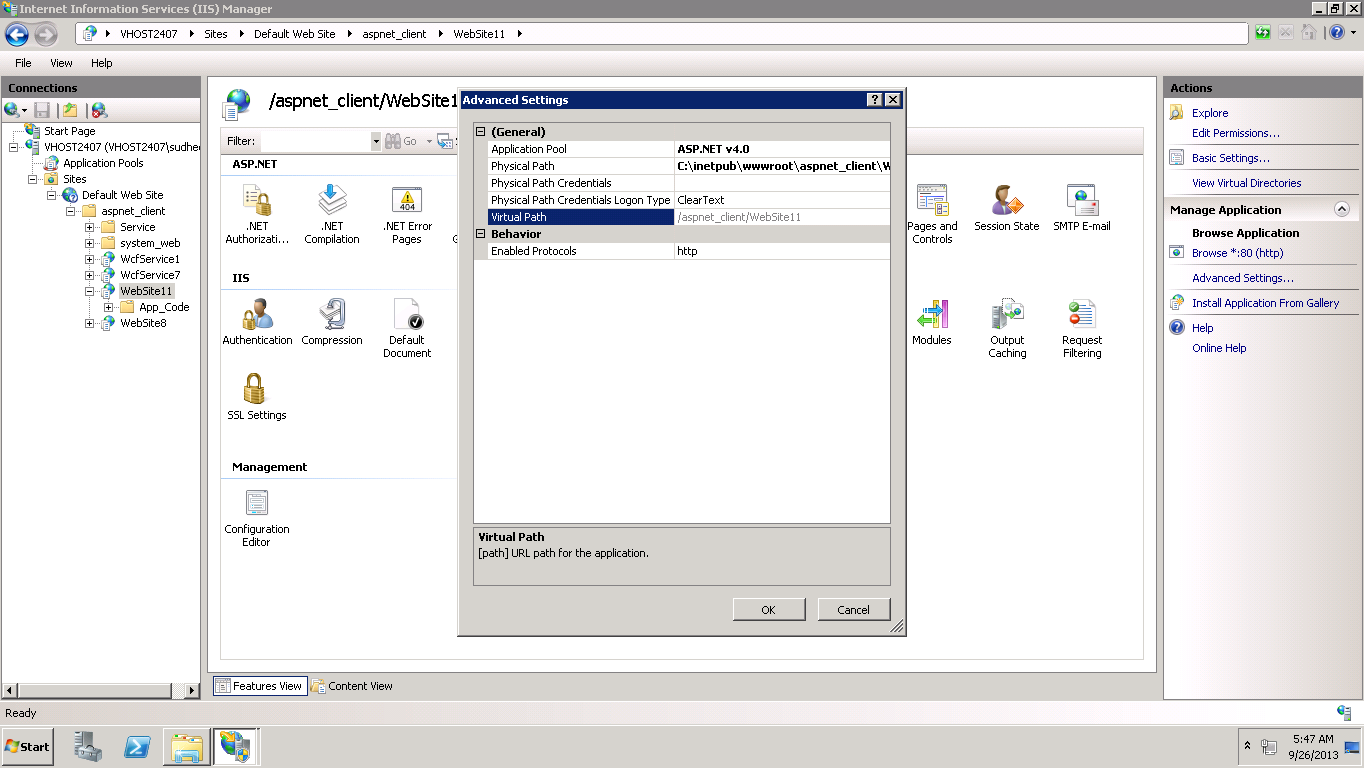
The file is copied into c->inetpub->wwwroot->ASP\_client



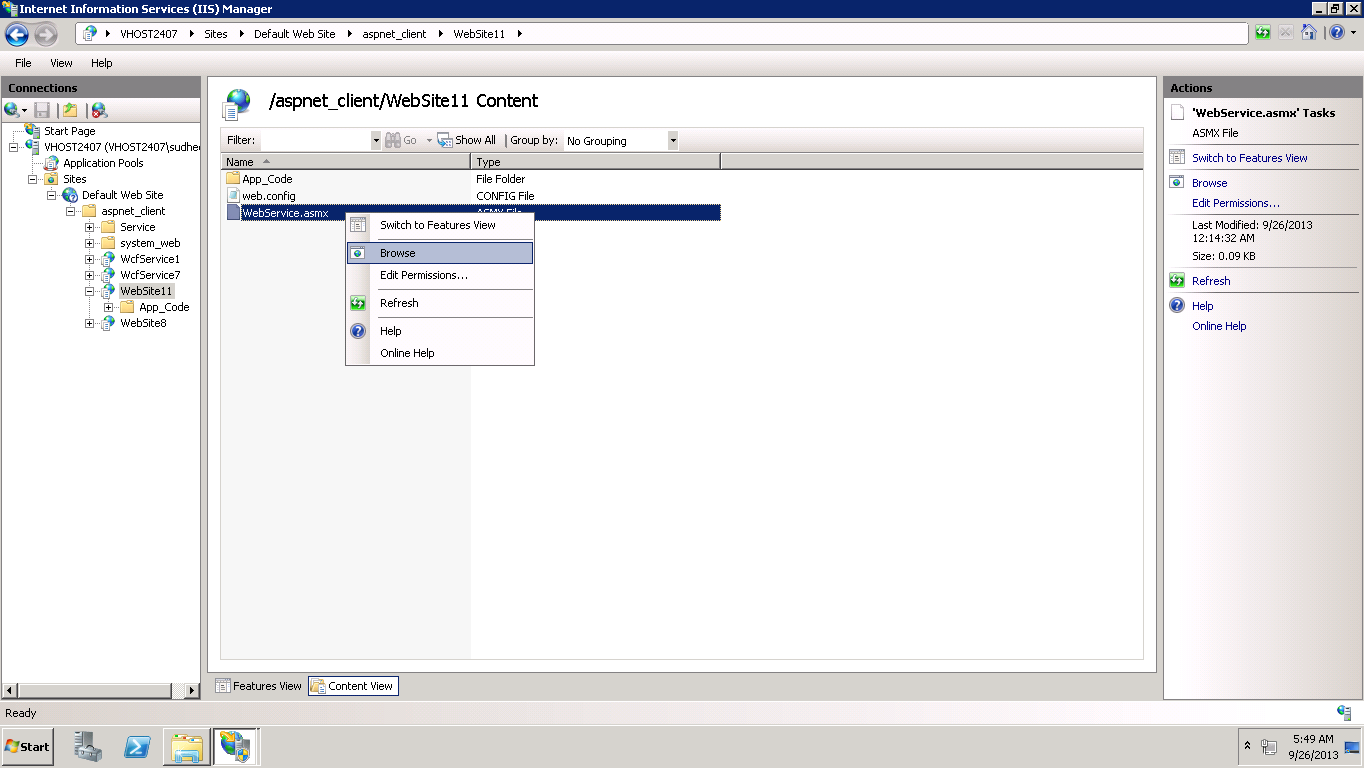
By clicking on the IIS manager and converting the file into the application, then the group ”IIS\_IUSRS” is added and access control is fully selected



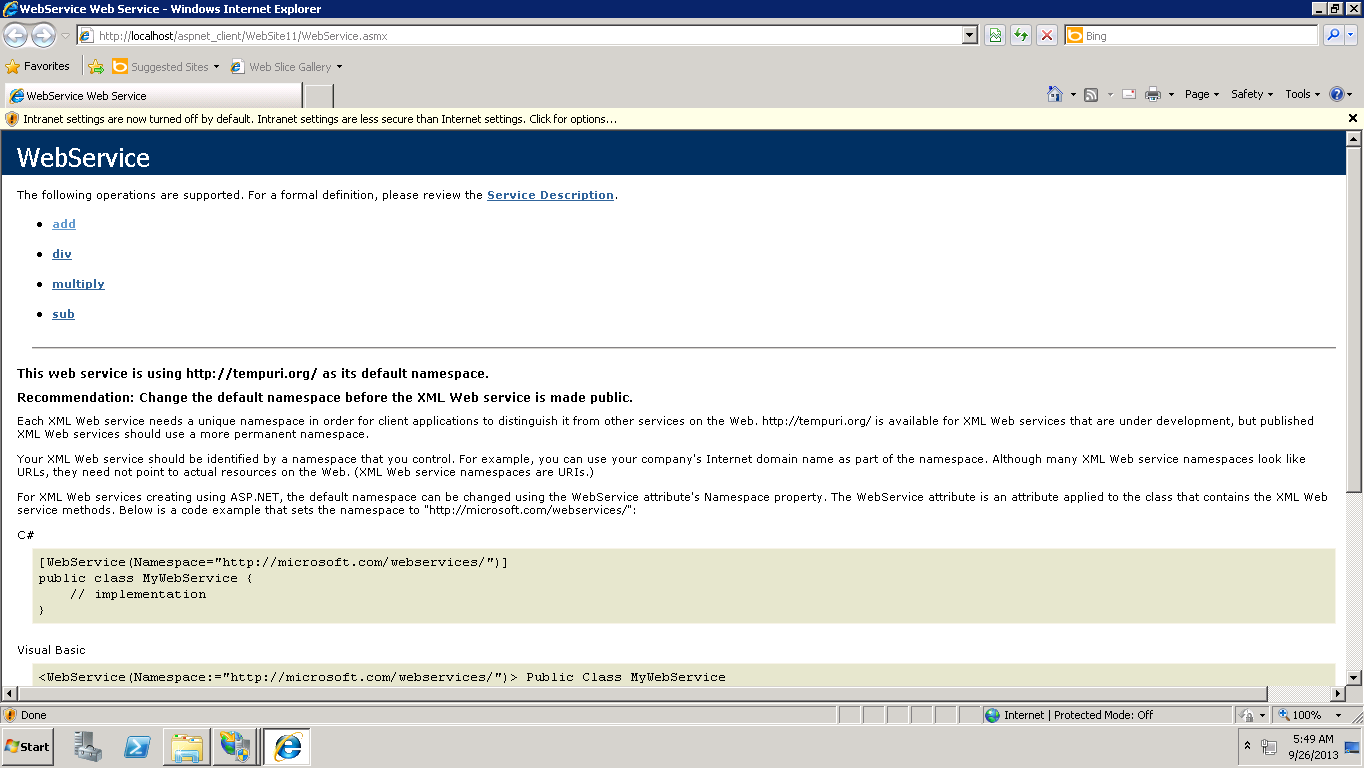
Change the application pool to ASP.NET v4.0 in the advanced settings



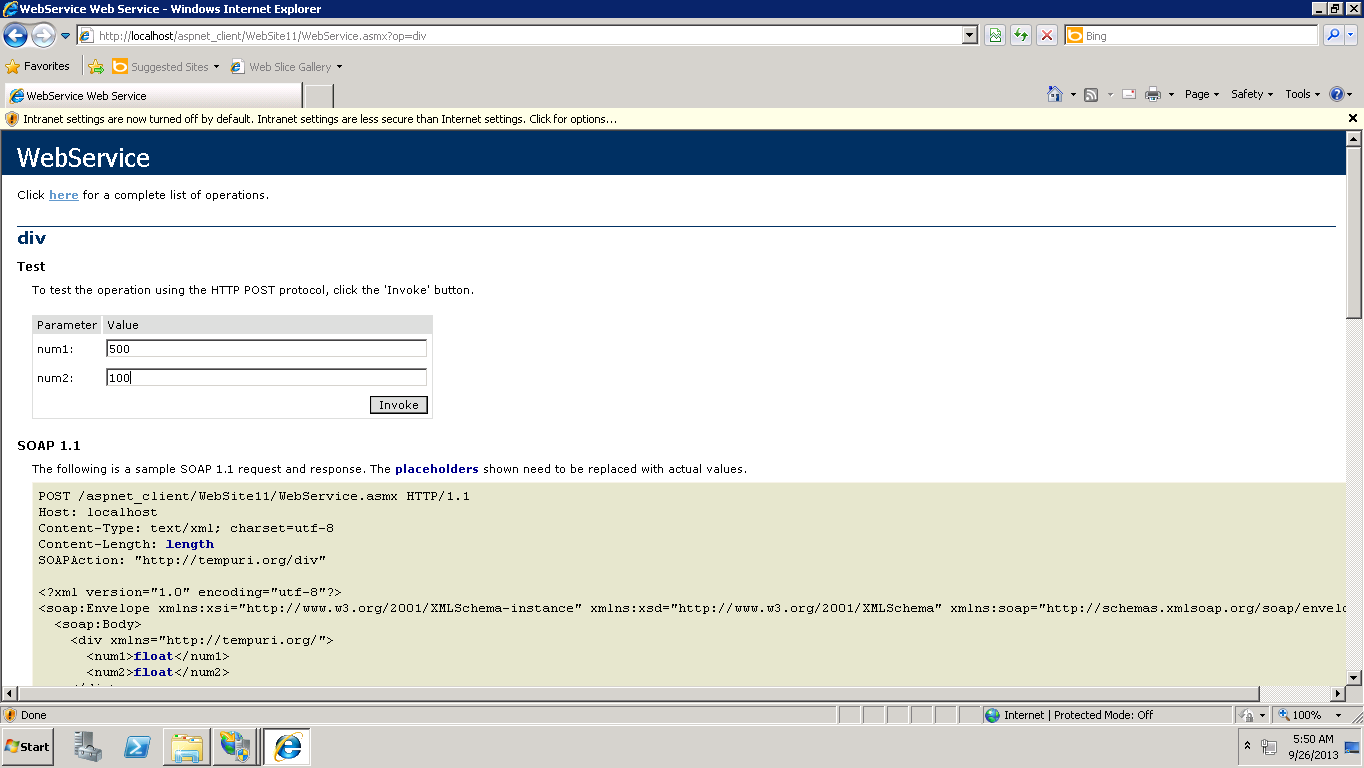
The application is opened in the content view and the webservice.asmx is right is clicked and browse is selected



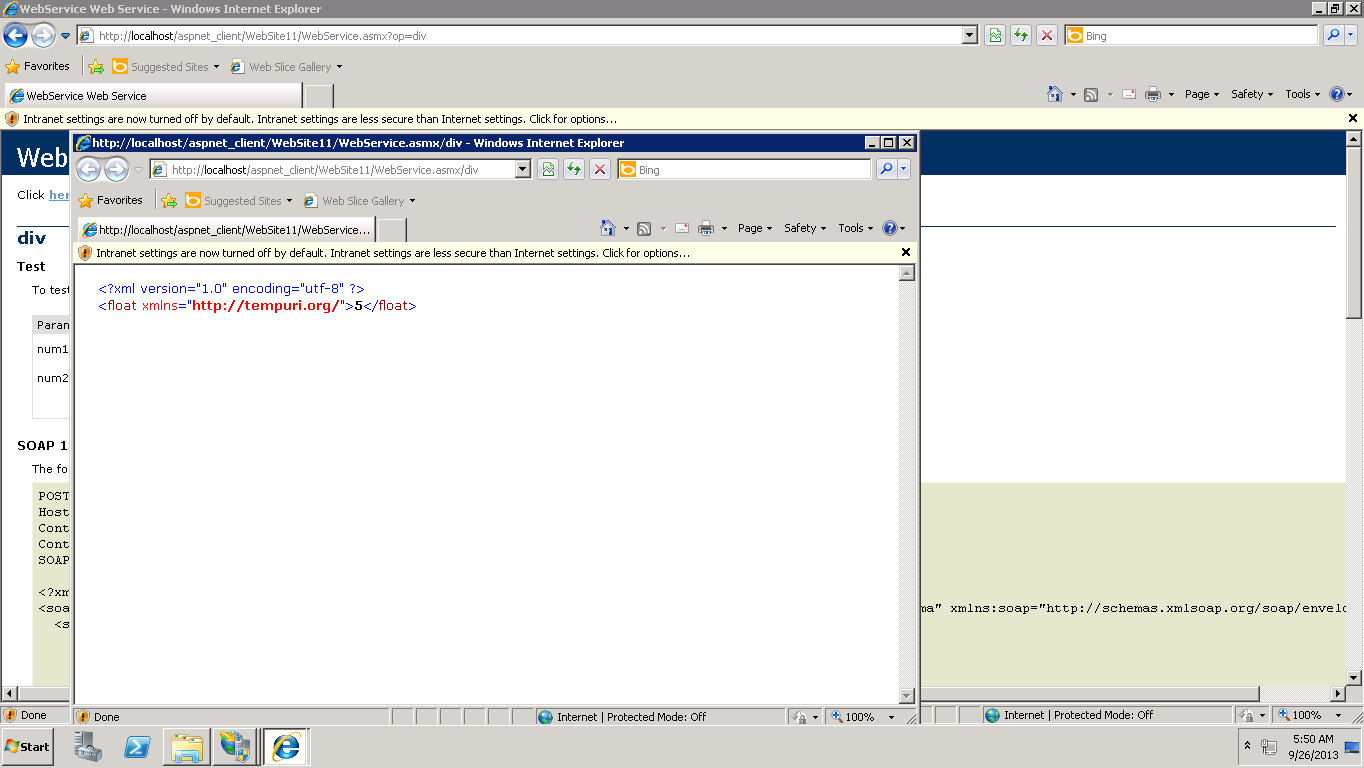
The desired output page is displayed



Division(div) operation is selected and the numbers are entered and the invoke button is clicked



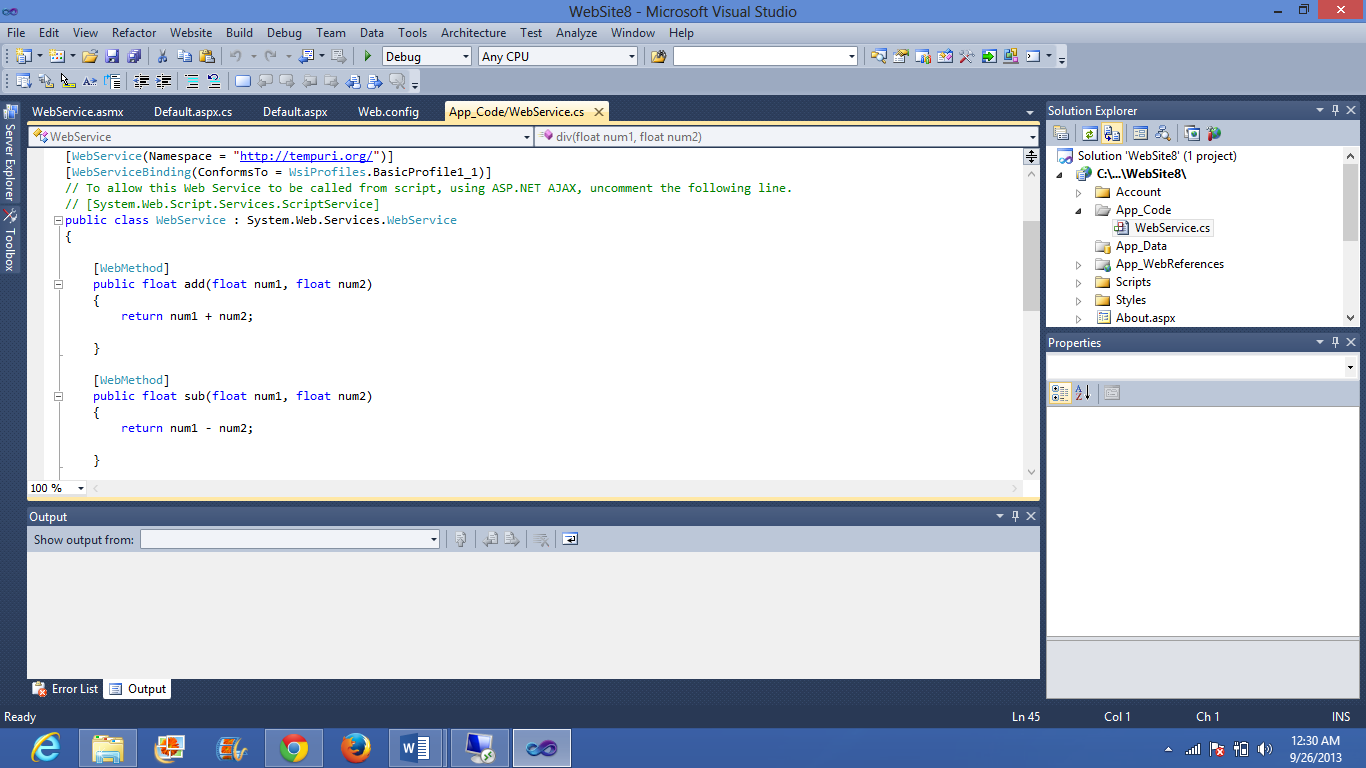
The following output is obtained.



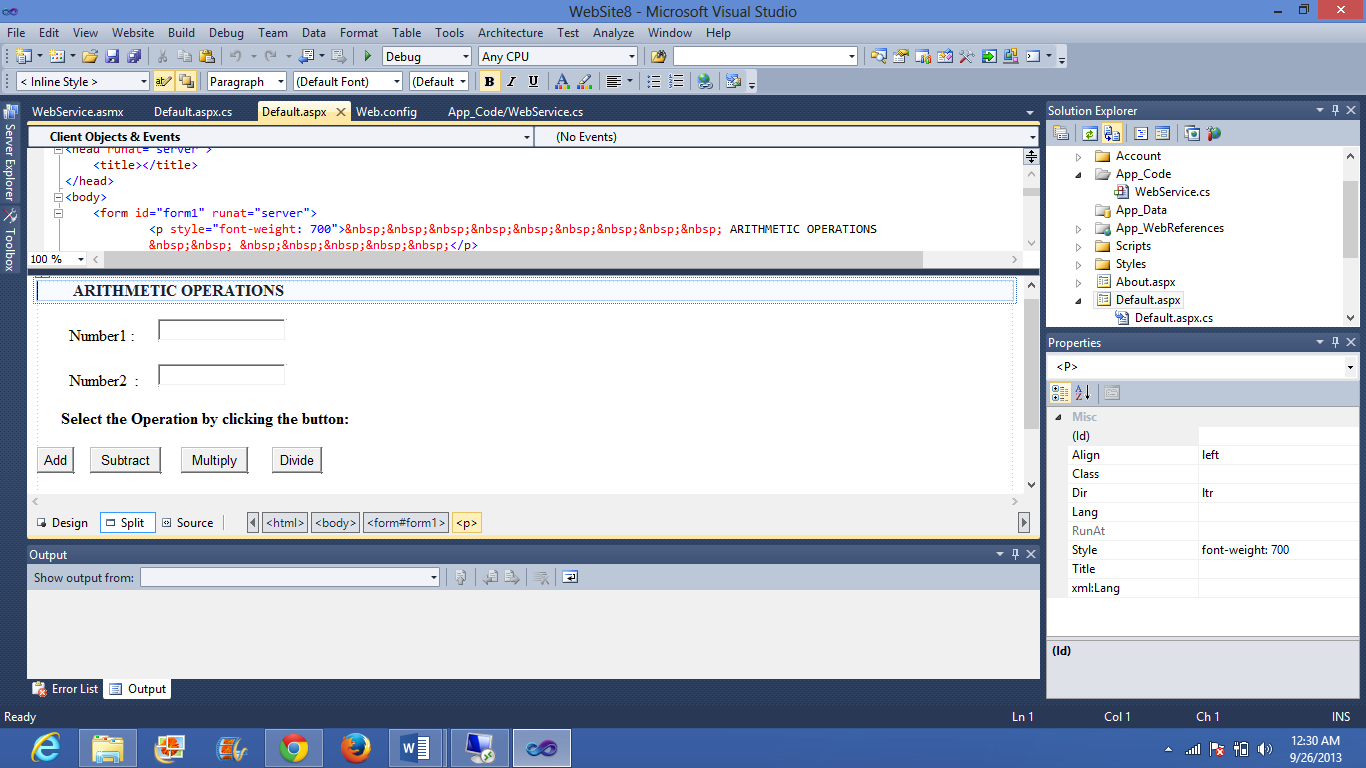
Hence, the SOAP based web service is successfully deployed in to the cloud.

**Creating a simple web client service for accessing the arithmetic operations using web services.**

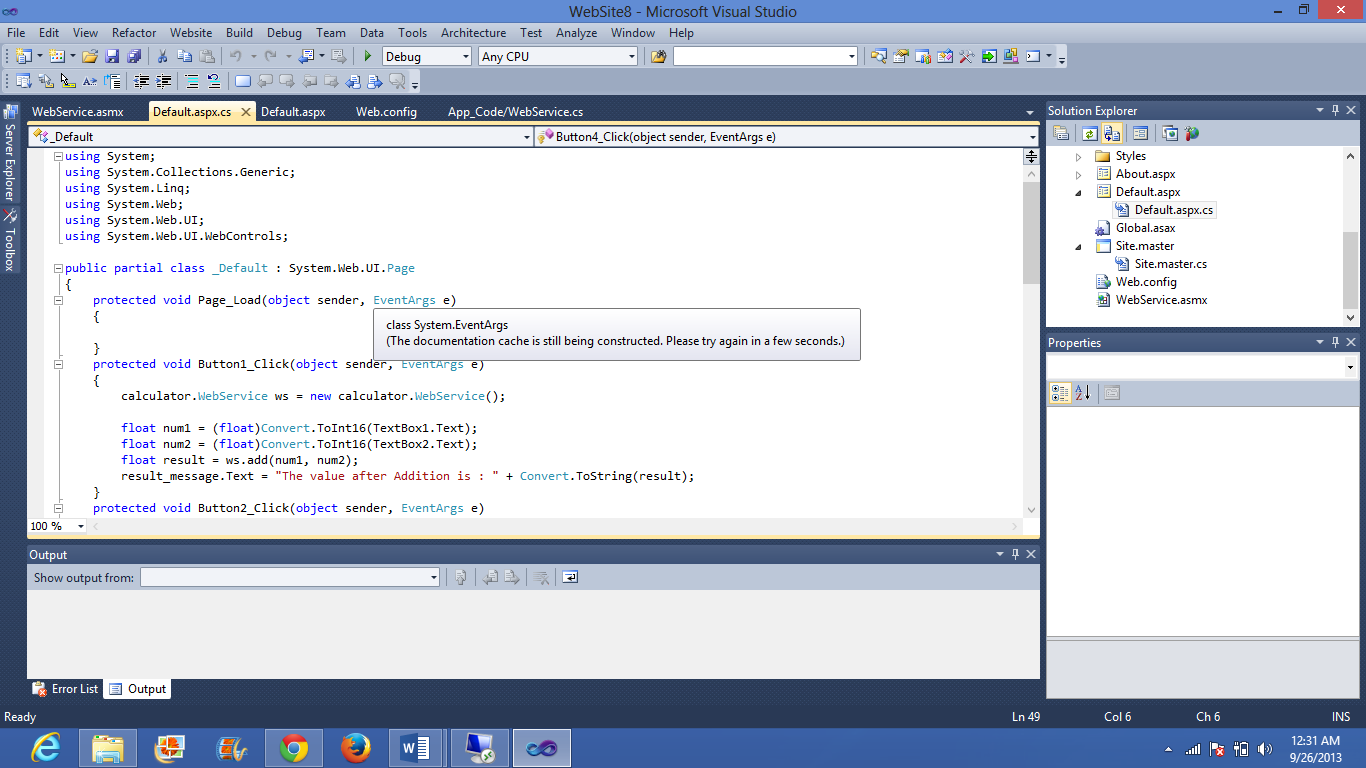
A new project is opened in the visual studio and Webservice .cs file is added to the website as the web reference.



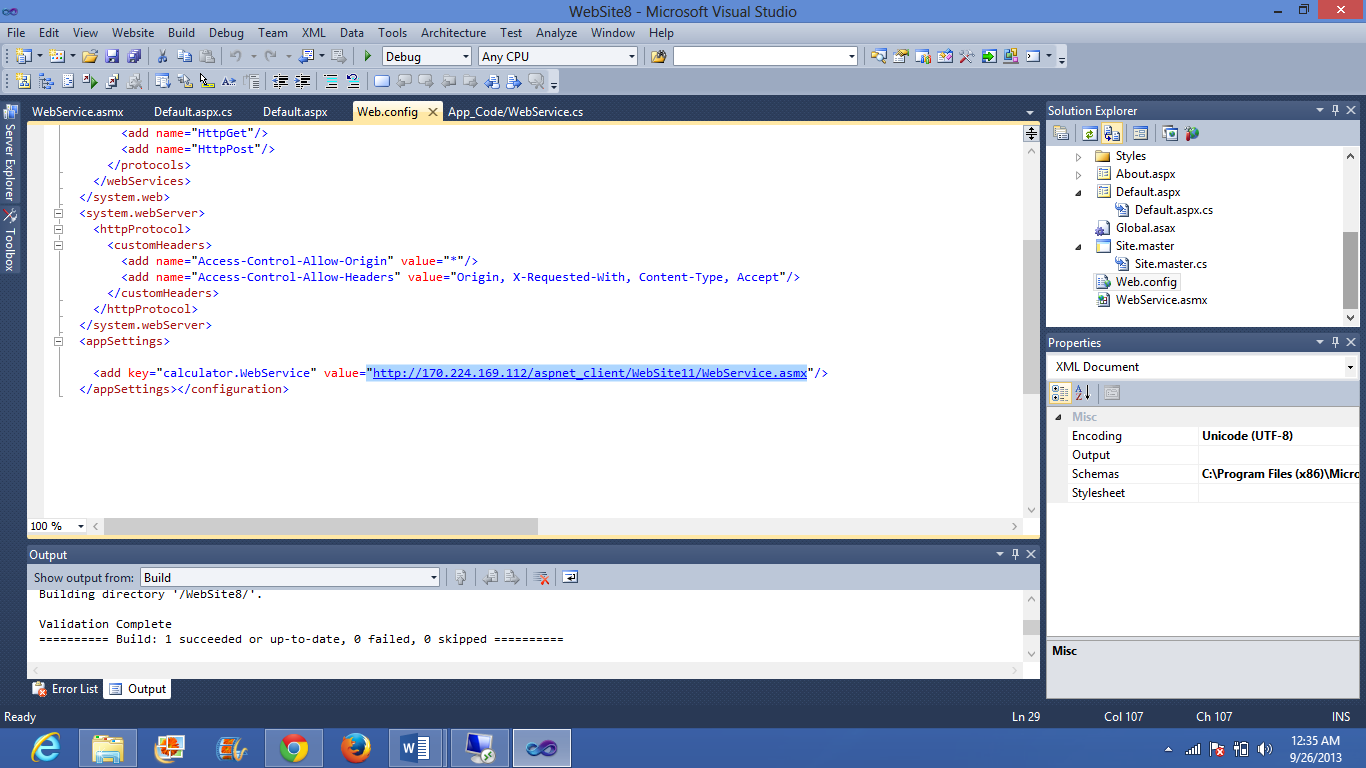
A web form is created as per the requirement



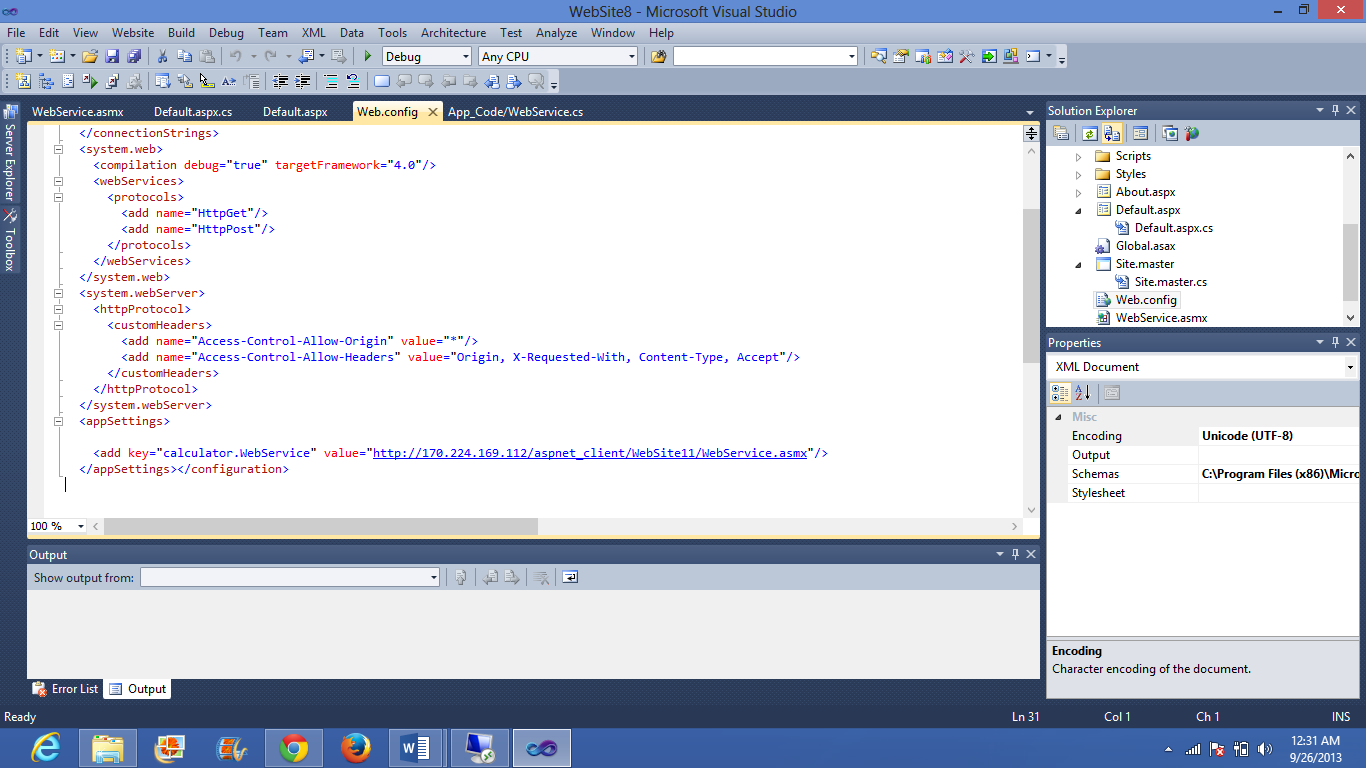
C# code is written in order to access the web form



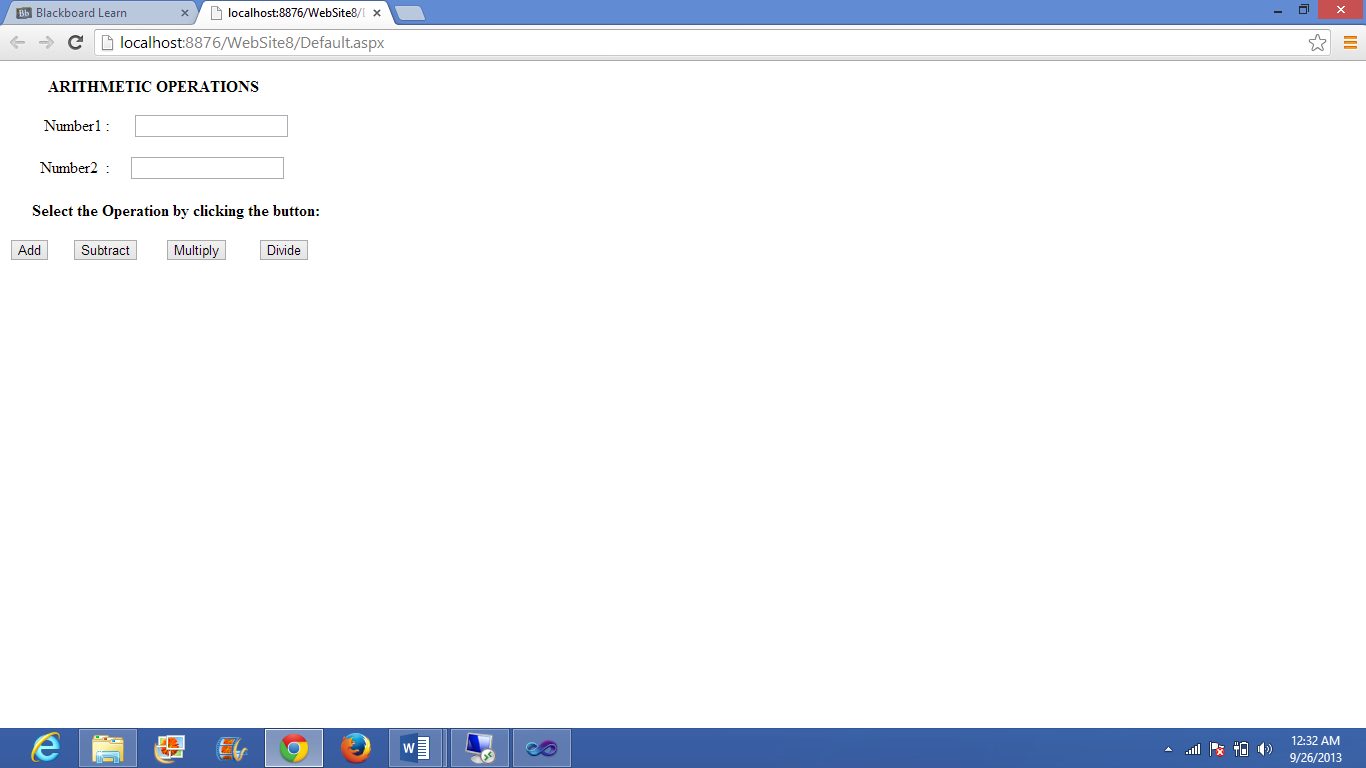
The link of the SOAP based service is added as the web reference



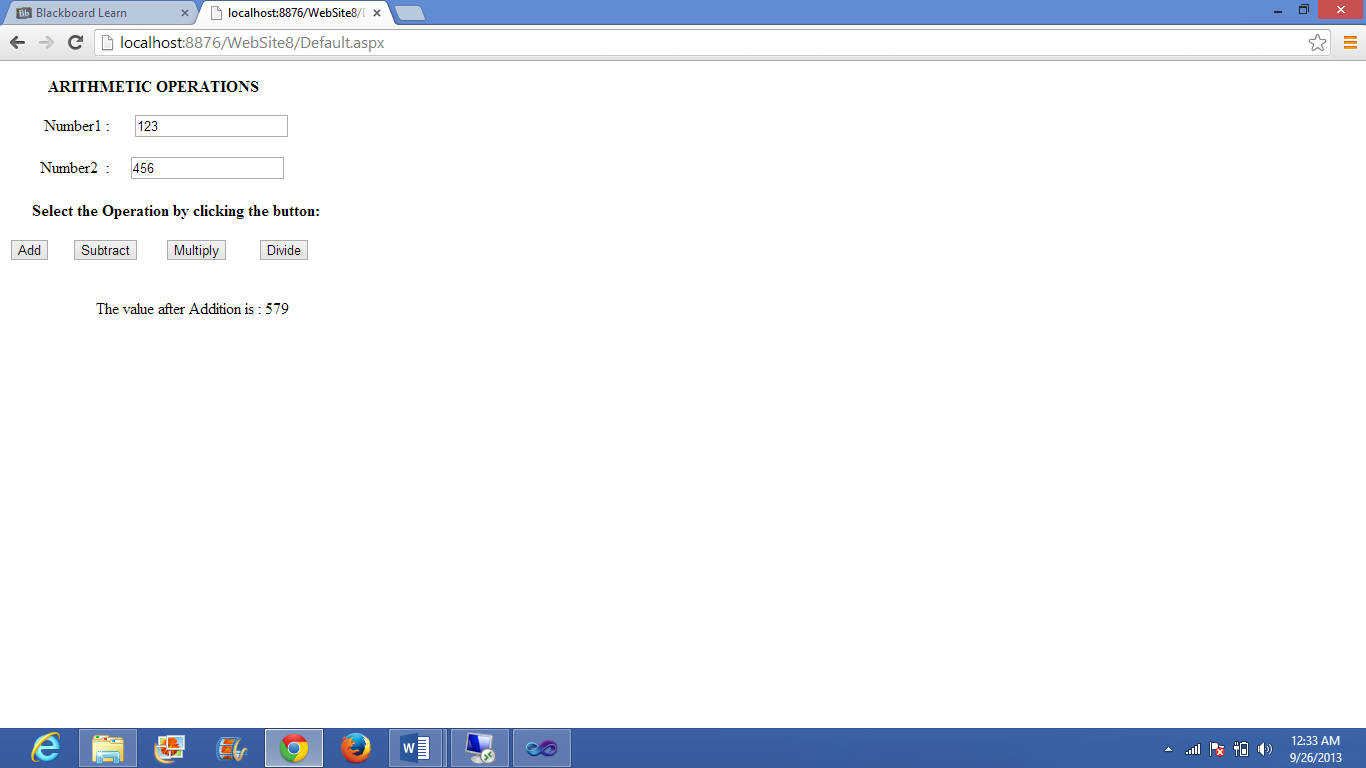
The web.config file looks like this after the web reference is added



Click on build and start without debugging the following web page appears



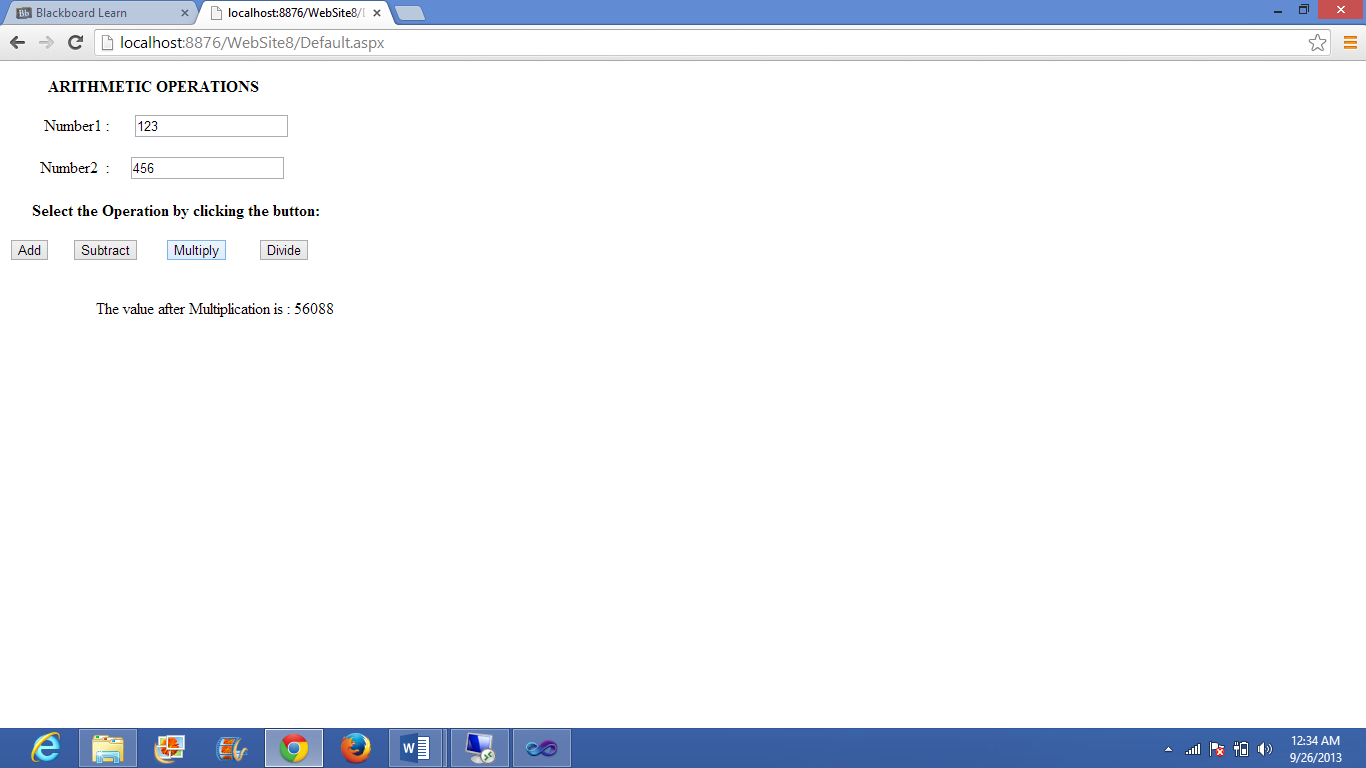
Output for addition



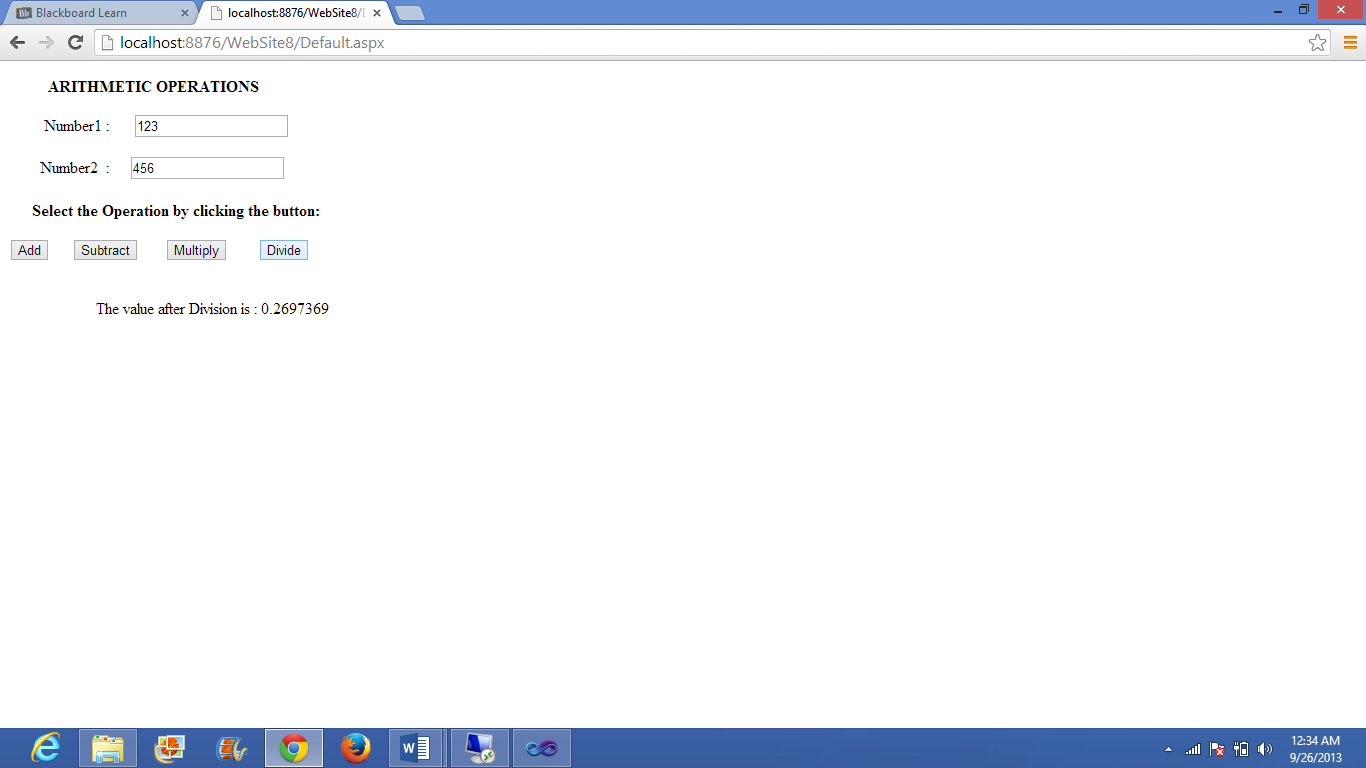
Output for subtraction



Output for multiplication



Output for division



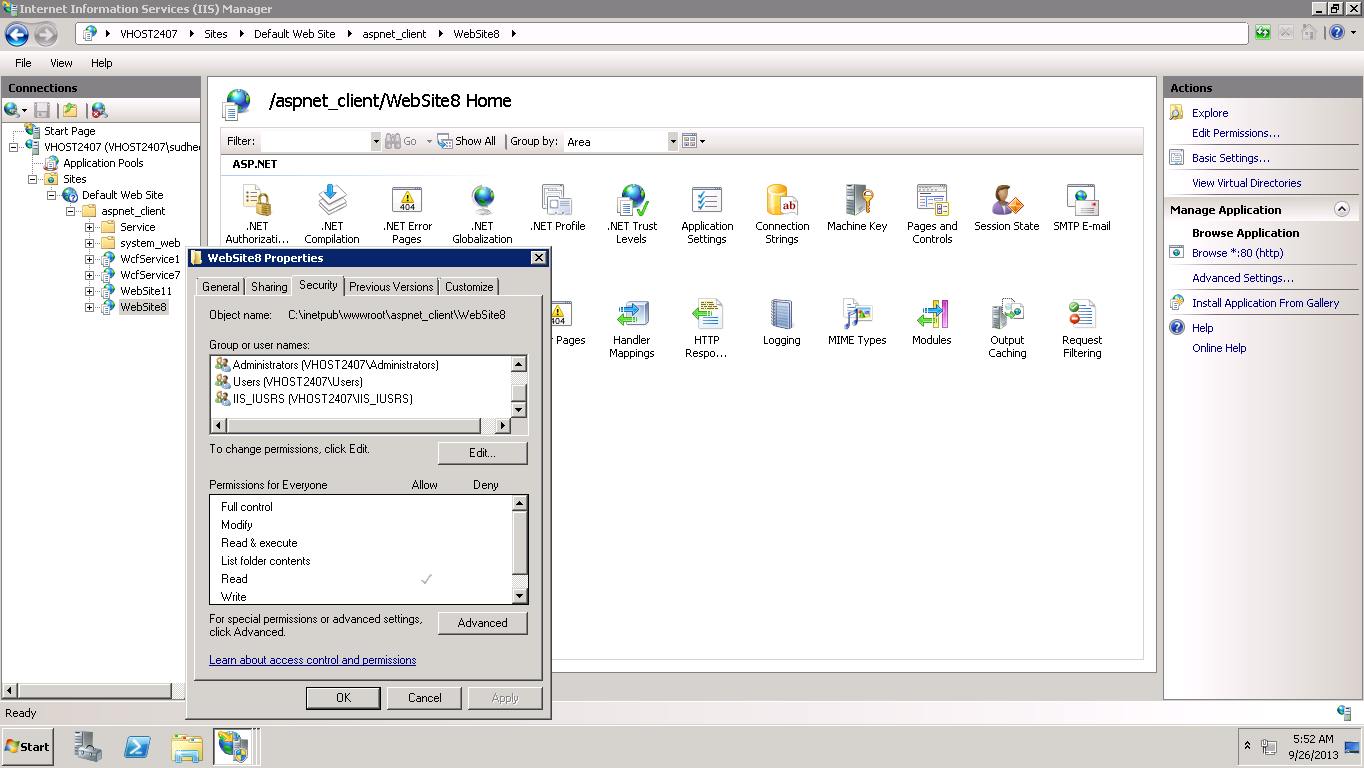
**Deploying the web client application in to the cloud:**

The website file is copied into the remote desktop

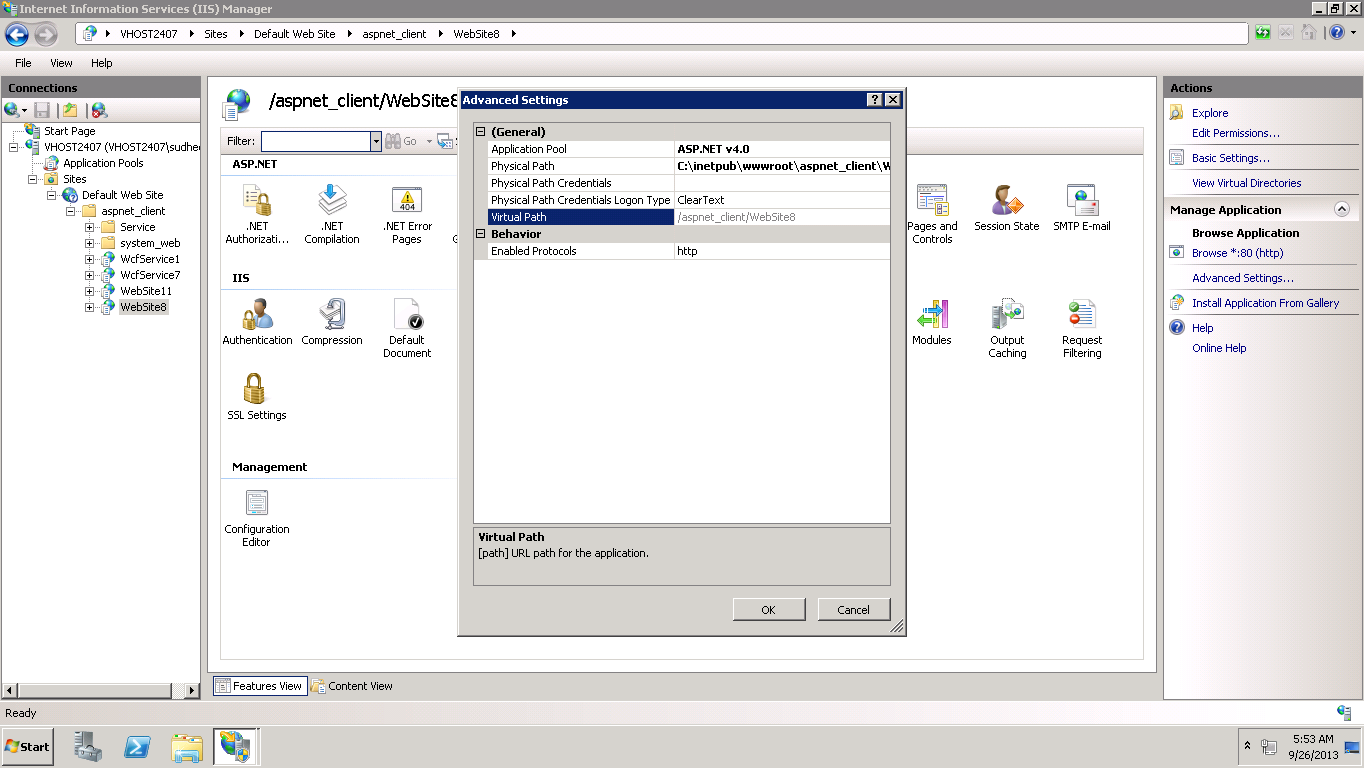


Then the file is copied into the c->inetpub->wwwroot->asp\_client

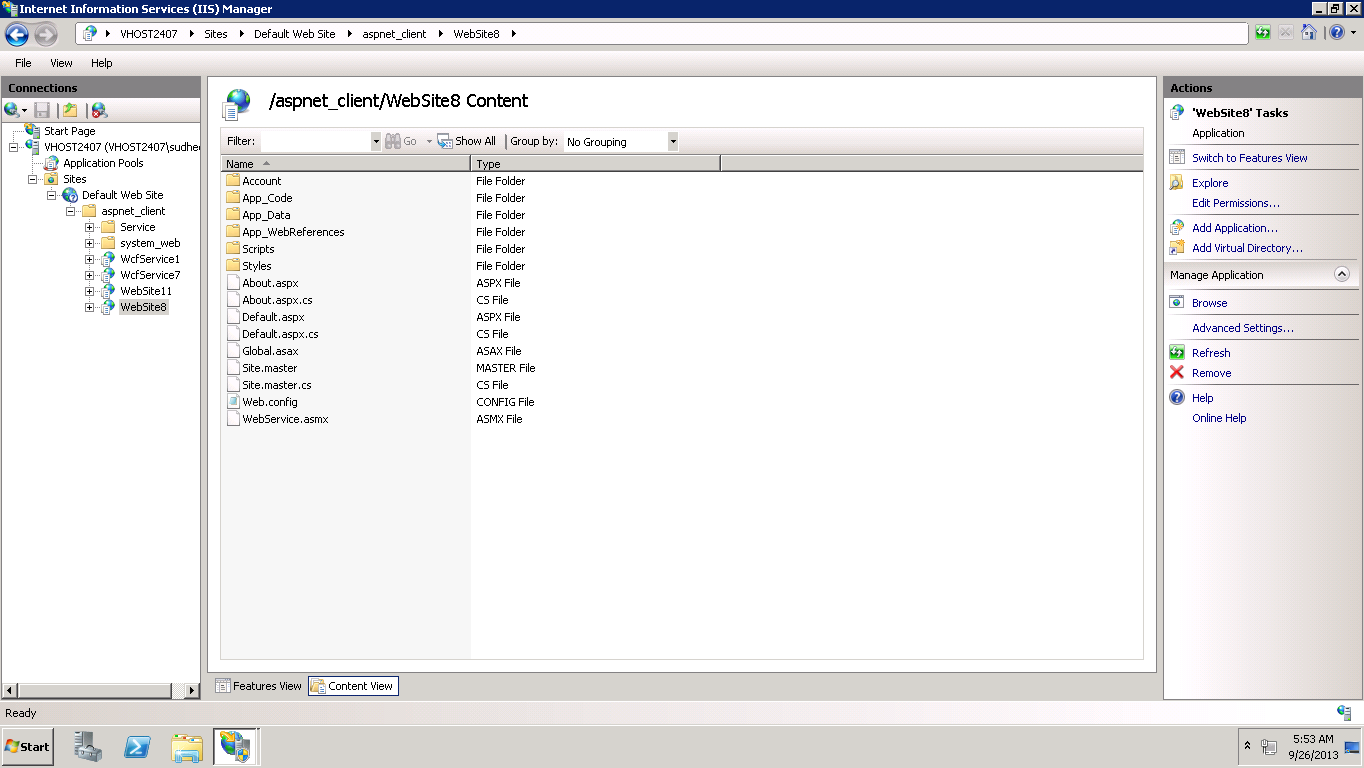
After opening the IIS manager the file is converted in to the application



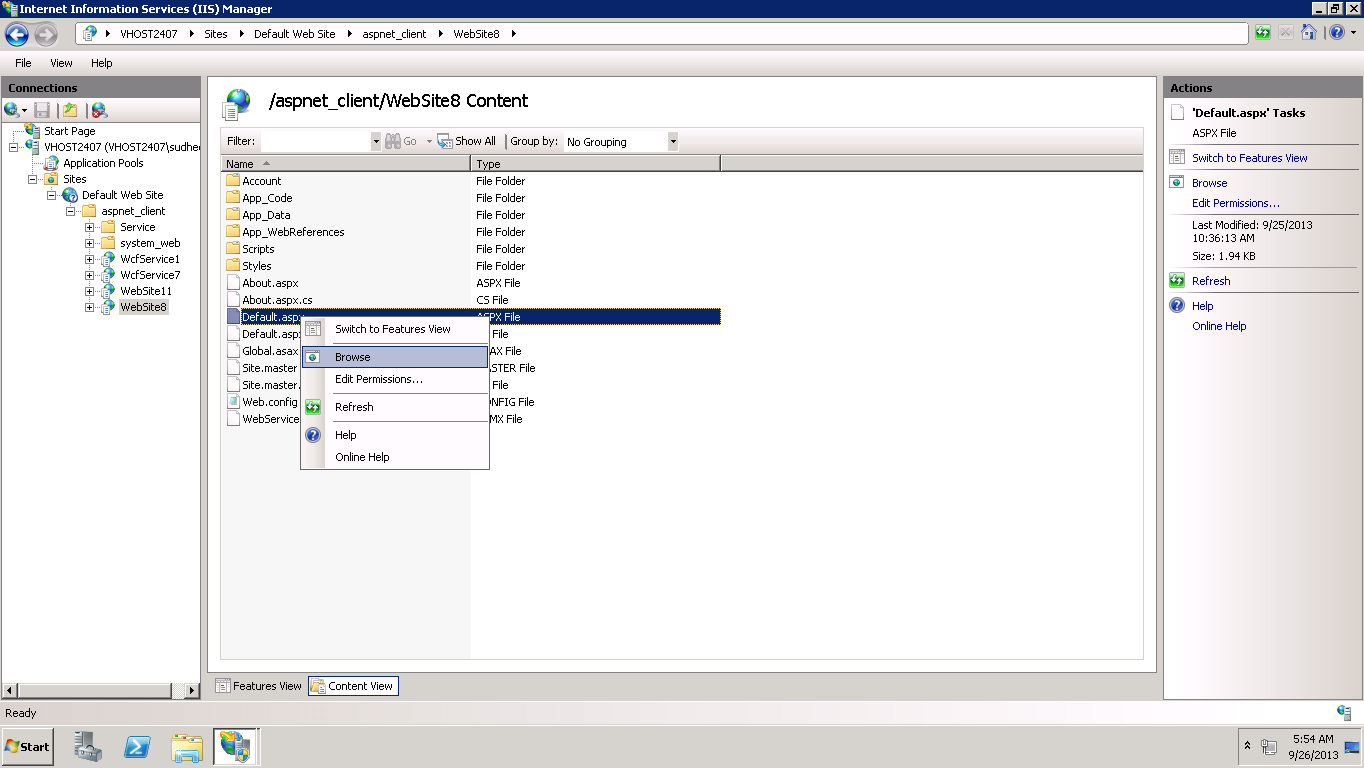
Changing the application pool to ASP.NET v4.0 in the advanced settings



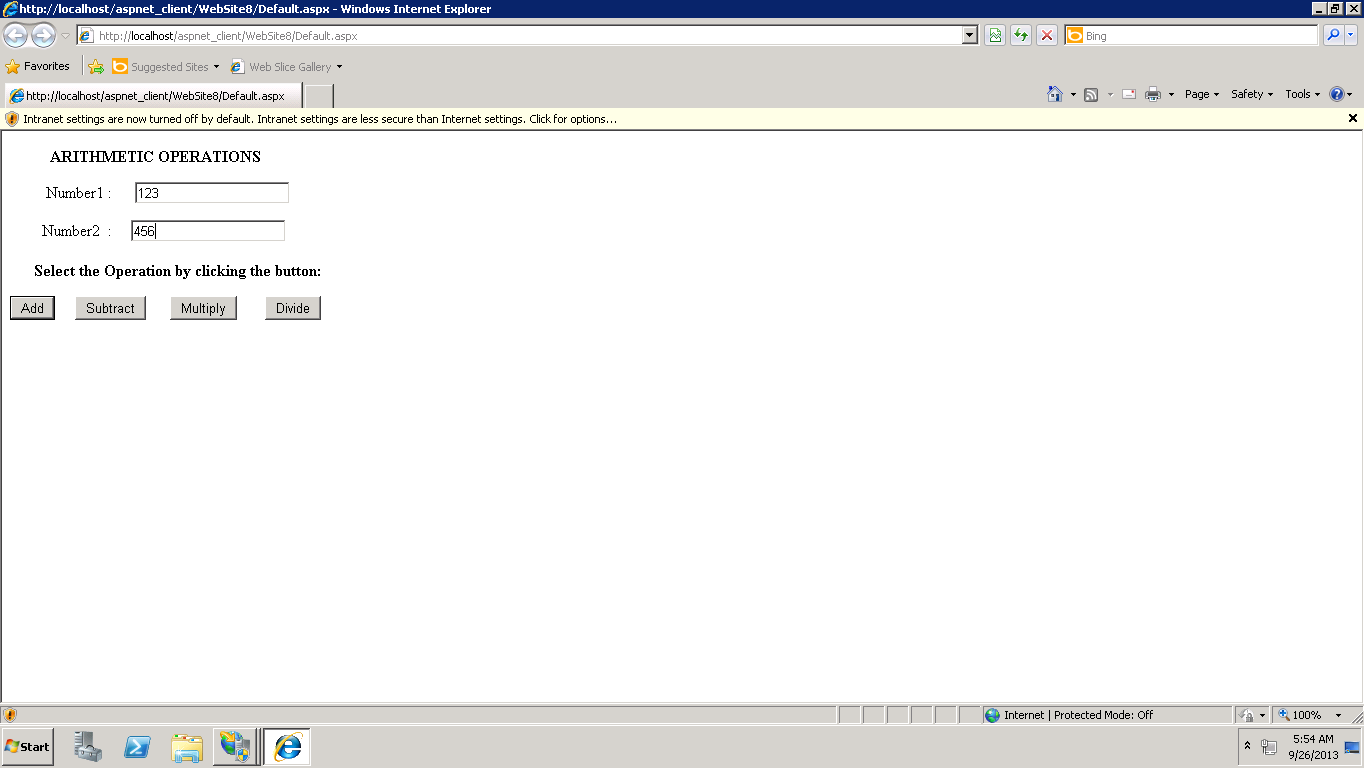
The application is opened in the content view



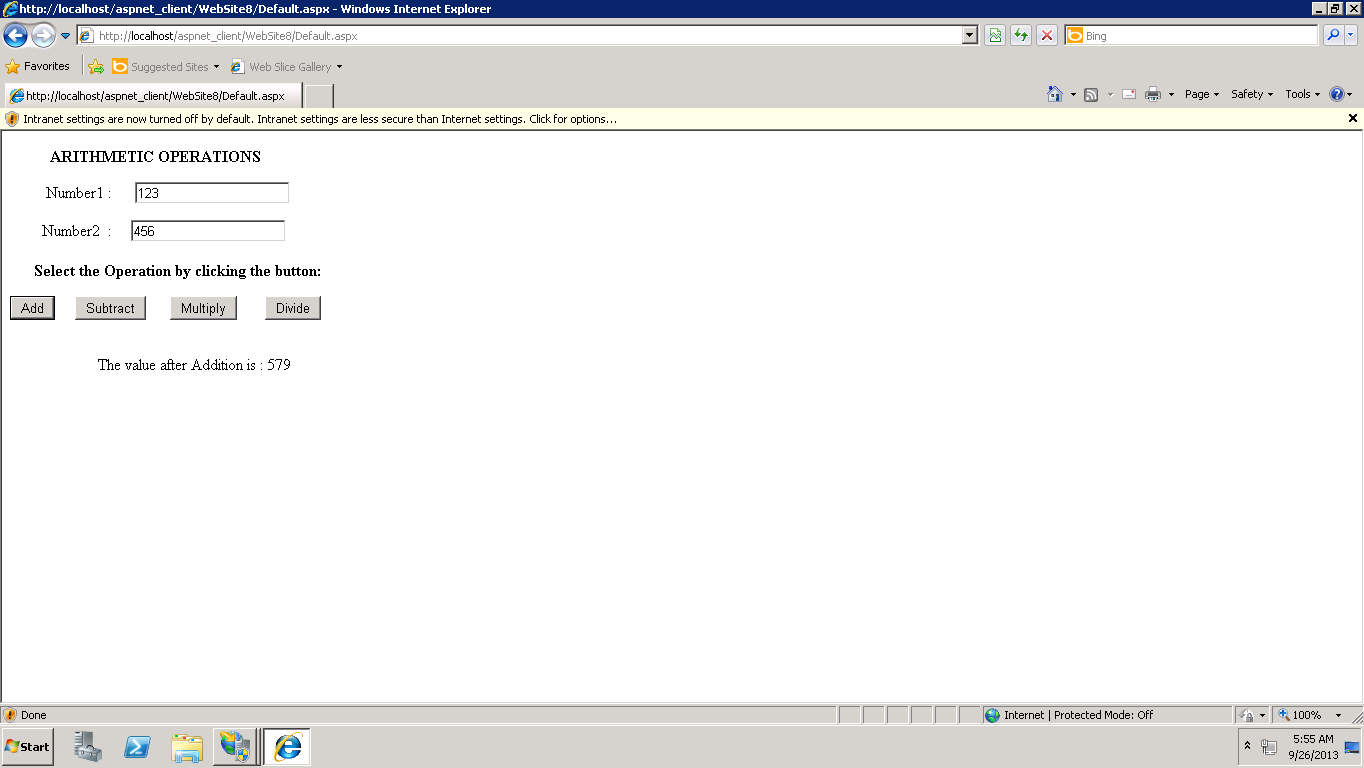
Right click on the default.aspx and click on the browse



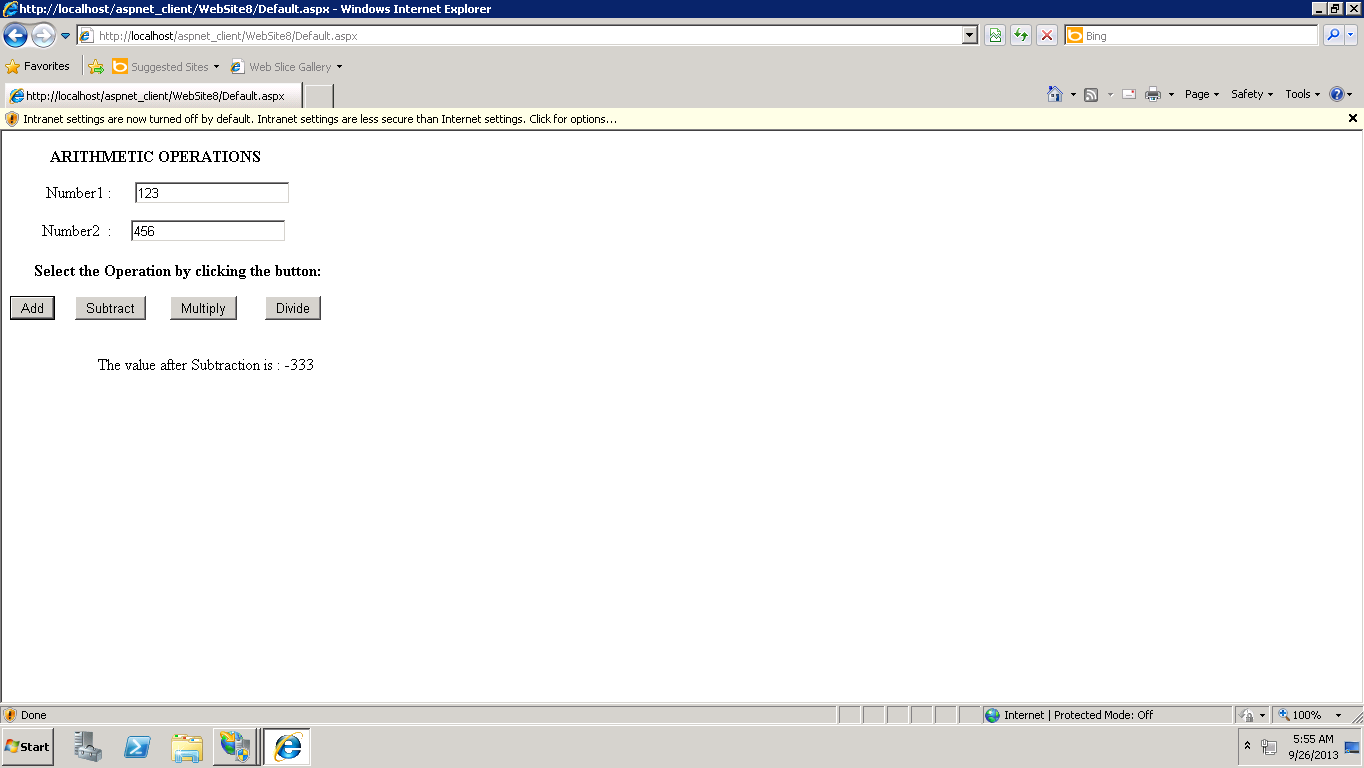
The following output page appears



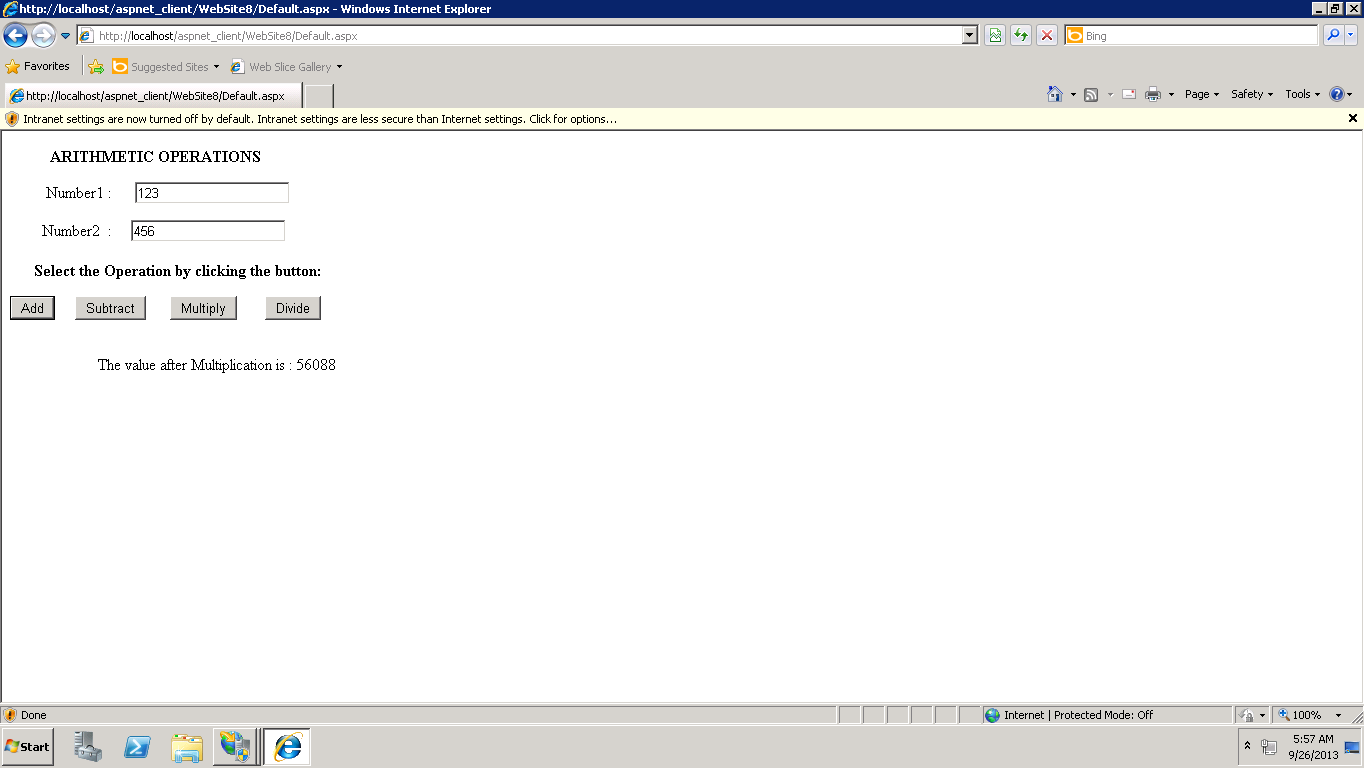
After entering the values and clicking on the add button the result displays

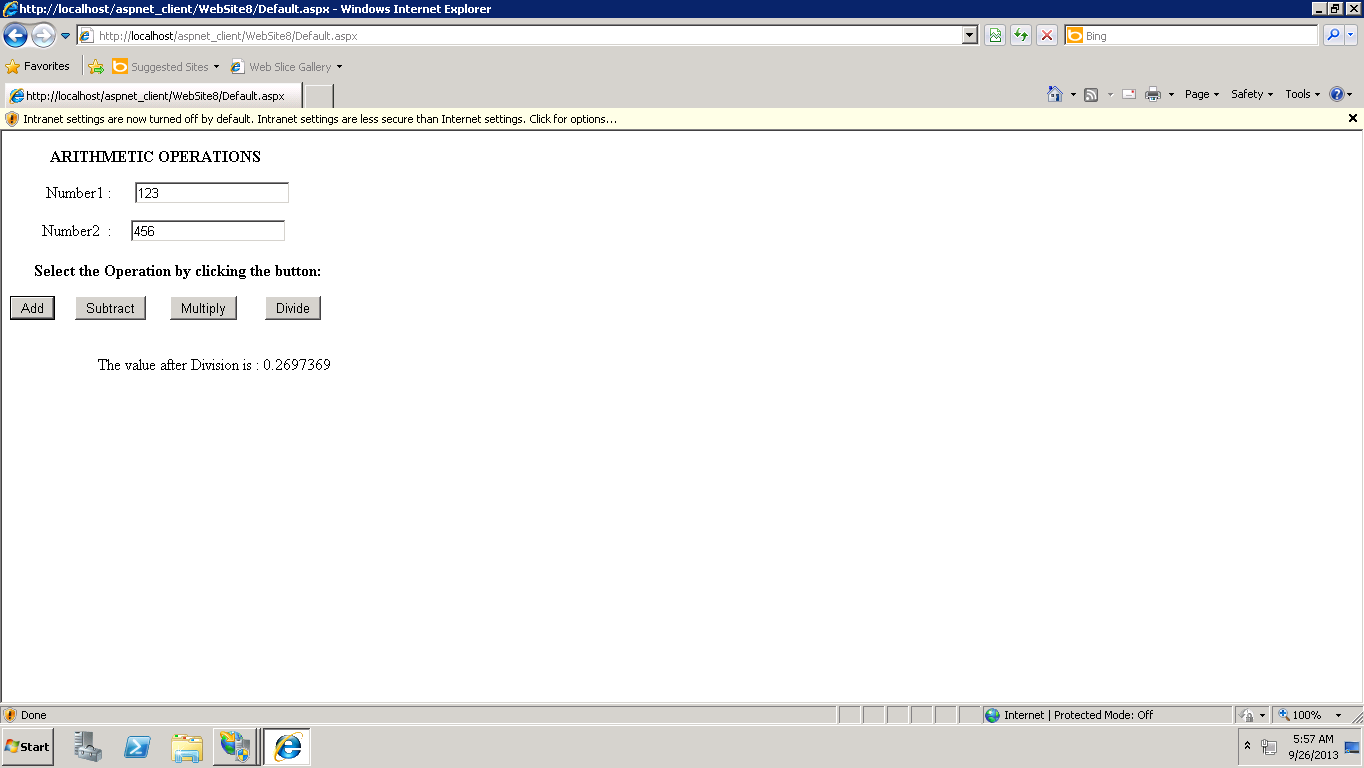


The result for subtraction



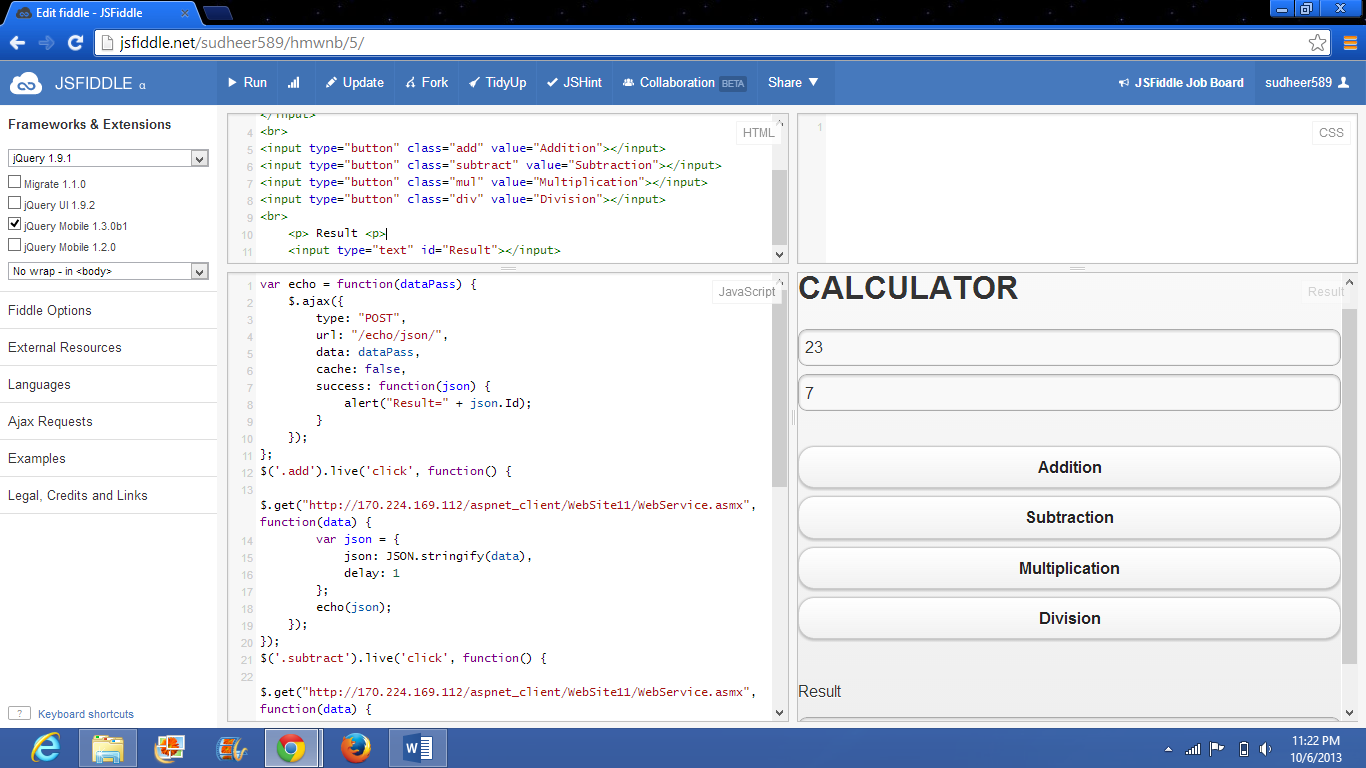
The result for multiplication



The result for division

Hence, the web client application using the SOAP based web service is successfully deployed into the cloud.

Web client jsfiddle



THANK YOU