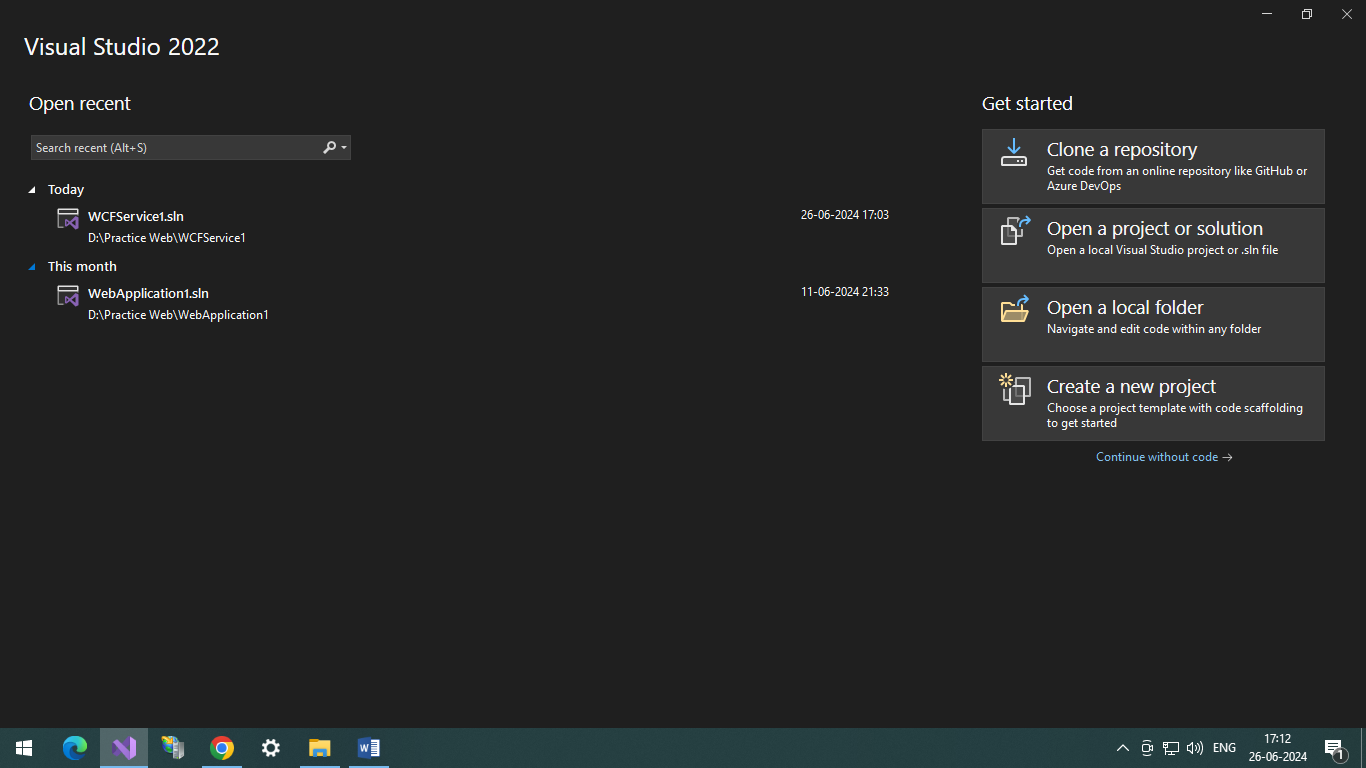
Refs:

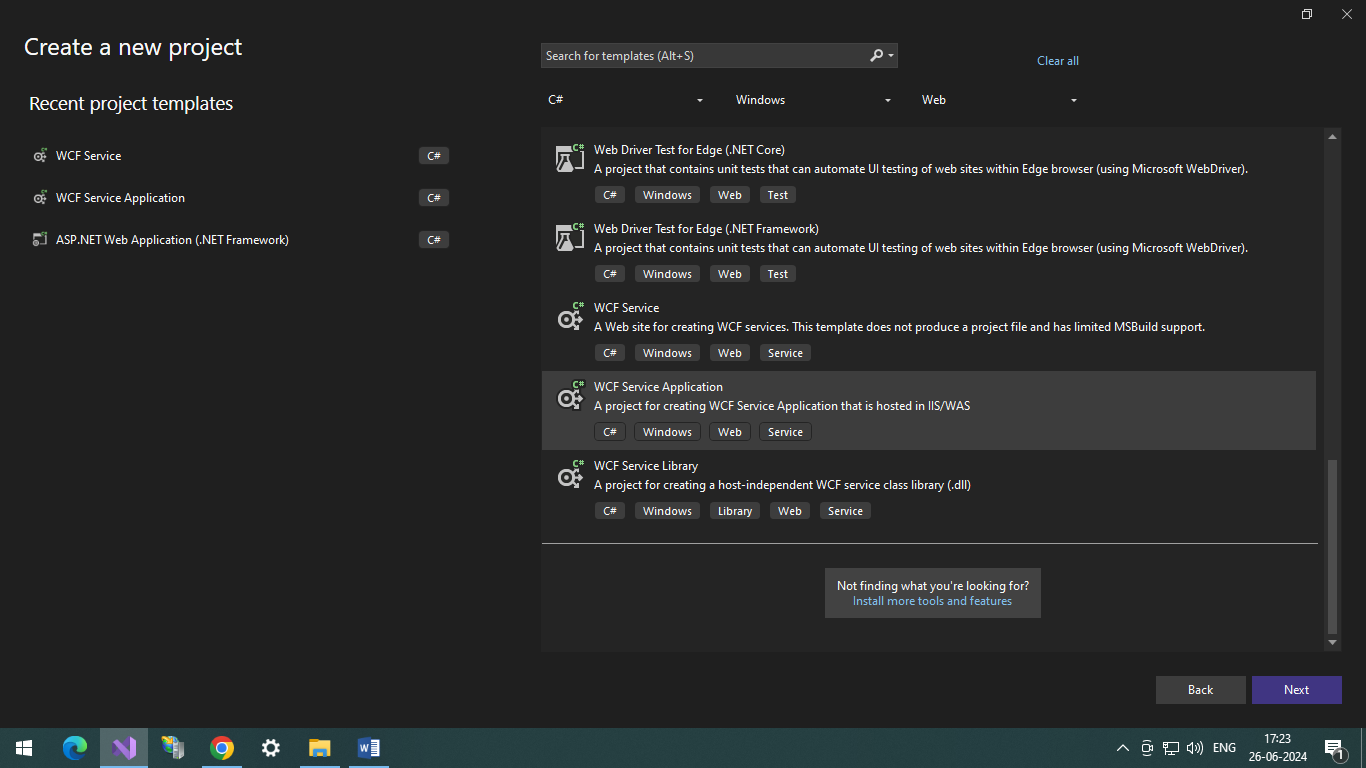
<https://www.c-sharpcorner.com/UploadFile/b182bf/introduction-to-wcf/>

<https://www.codeproject.com/Articles/201901/CREATE-RESTful-WCF-Service-API-Using-POST-Step-By>

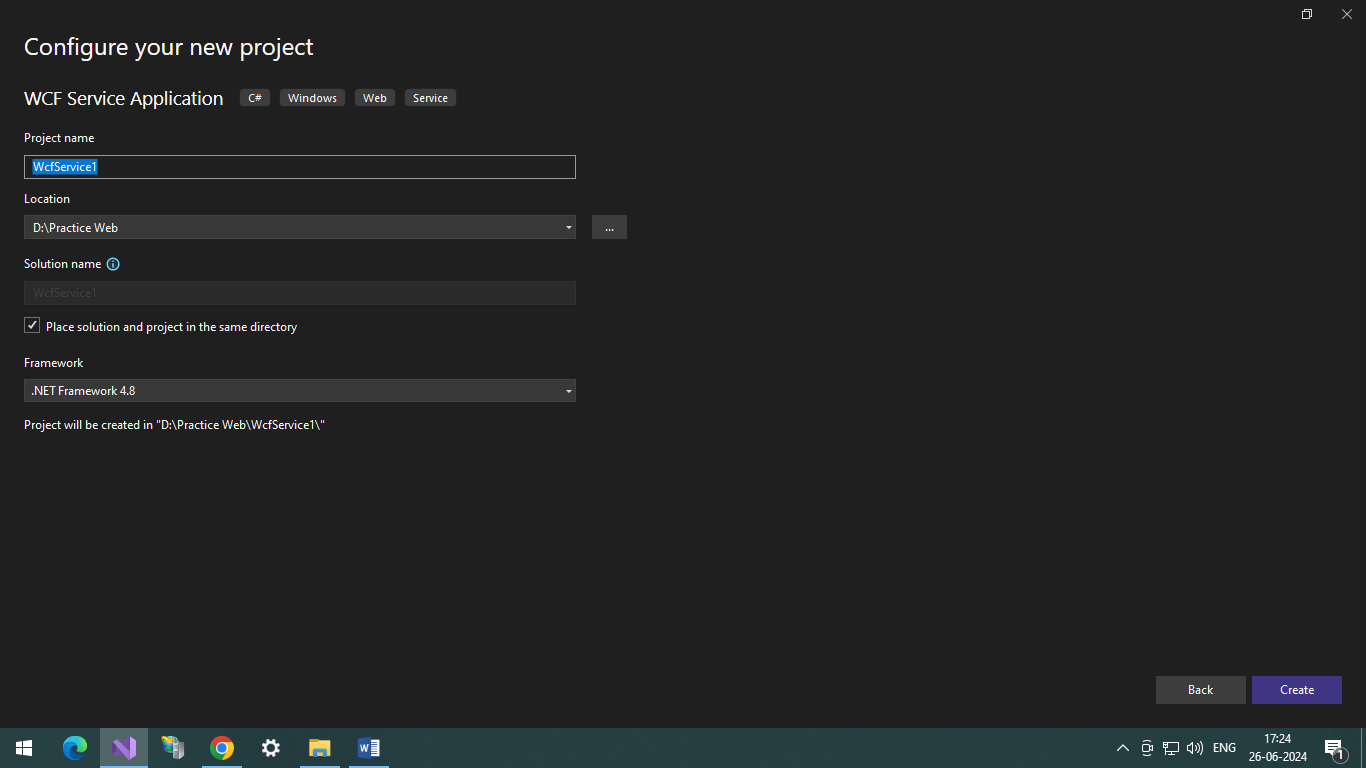
1. Create a new project



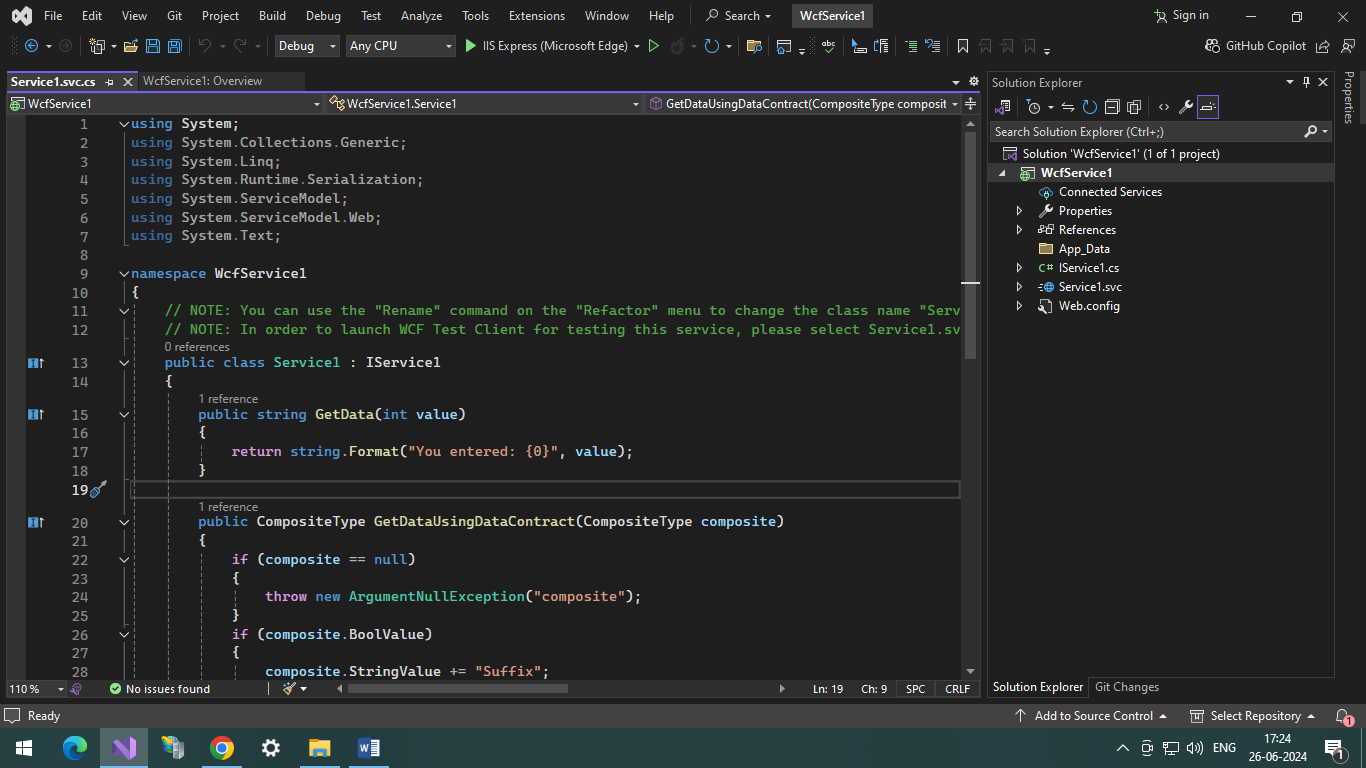
1. Select WCF Service Application and click next



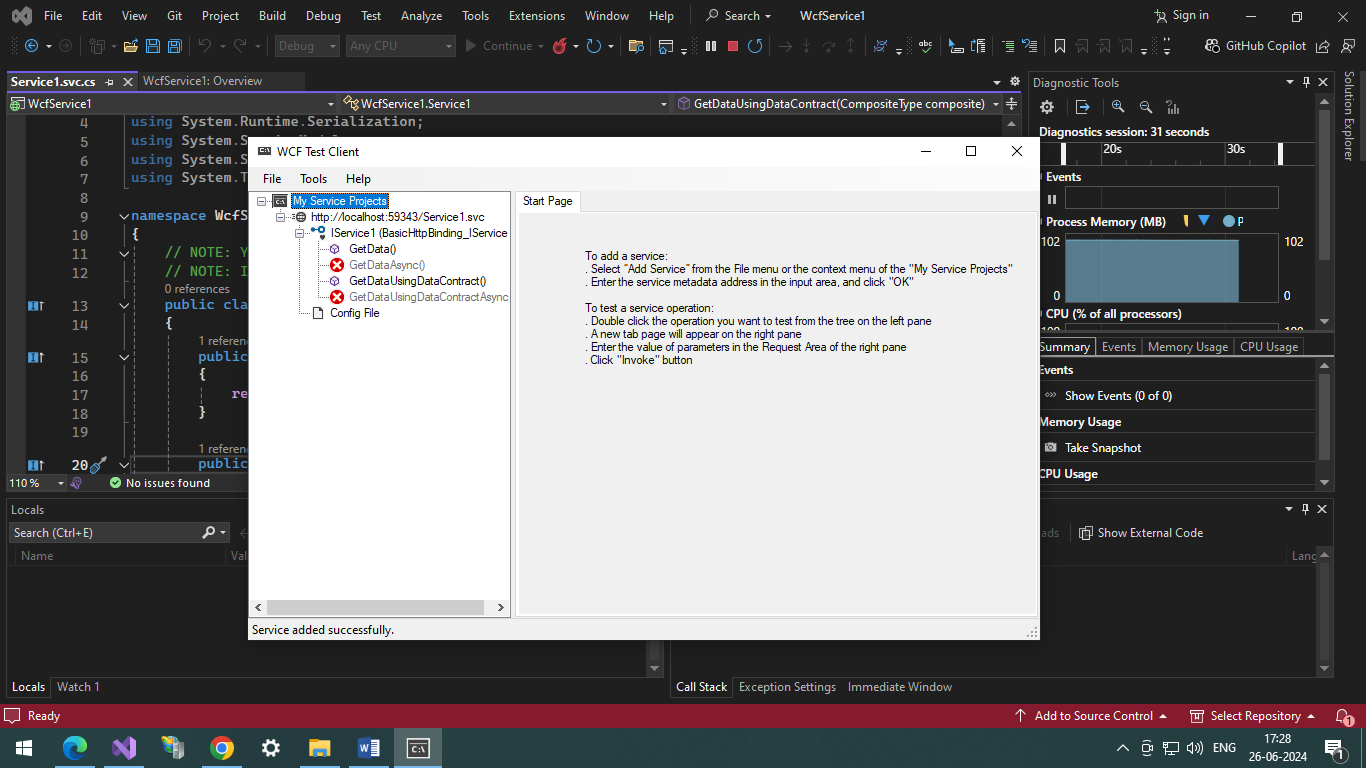
1. Give Project Name , Project path, checkbox checked and framework selected and click Create button.

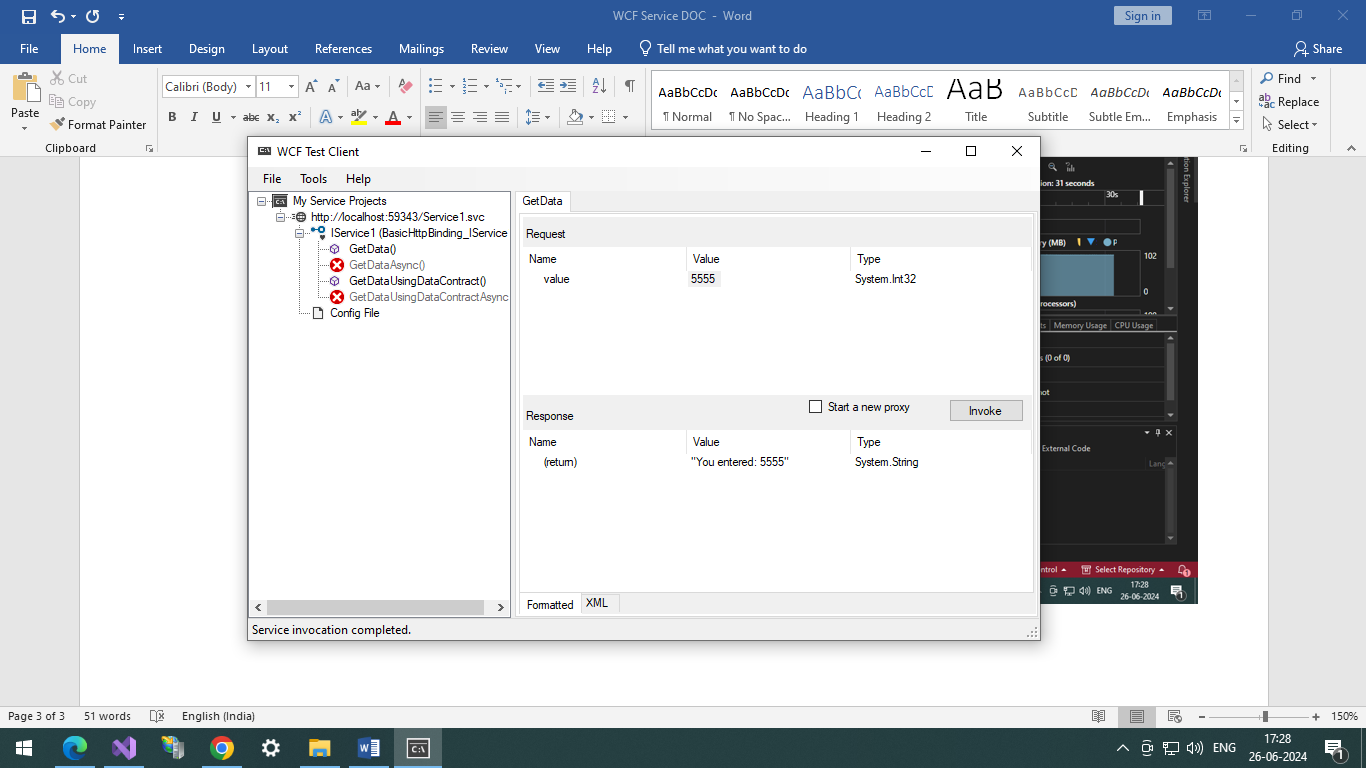


1. Service will be create with the following structure.



1. Once you Run the project it will start like this. You can test you service method which you have already declared in the service1.svc.cs class



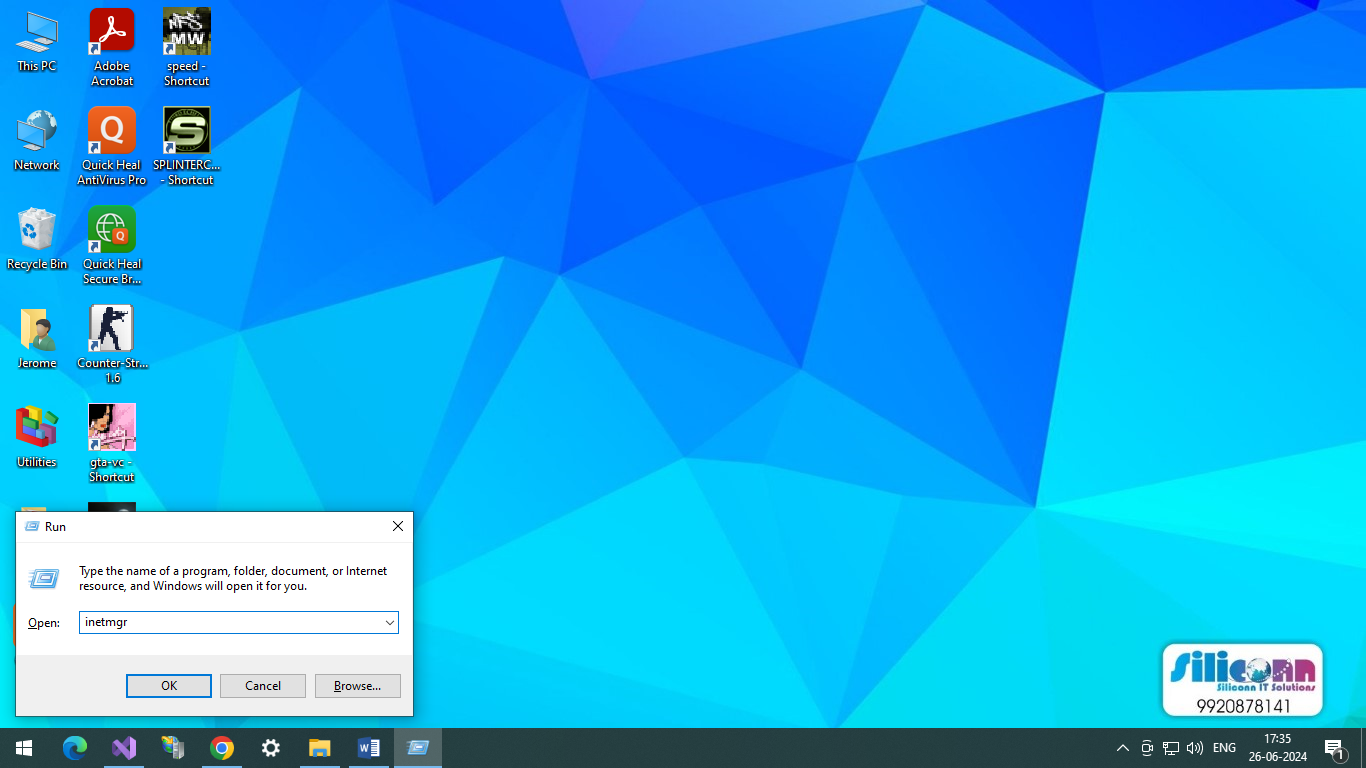


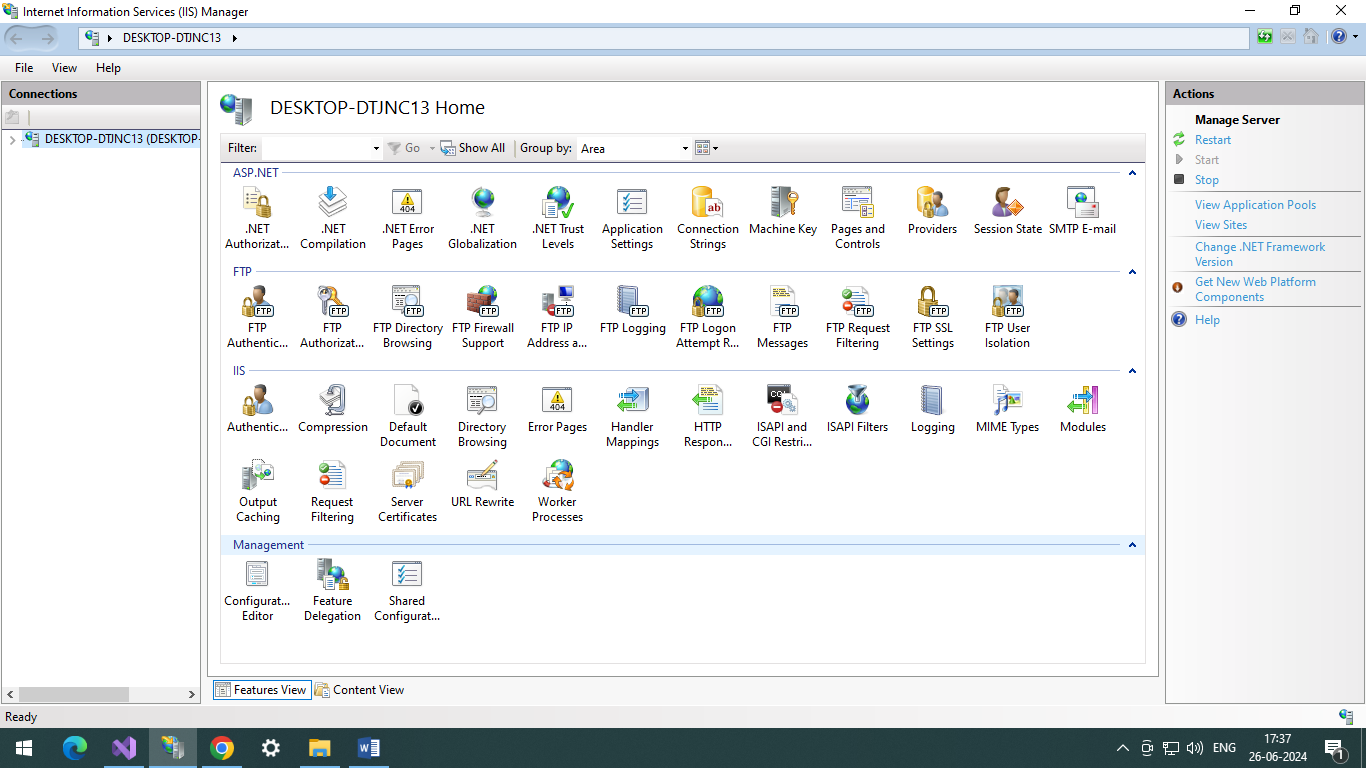
To host wcf service in IIS server.

1. Install the IIS server in the windows machine.

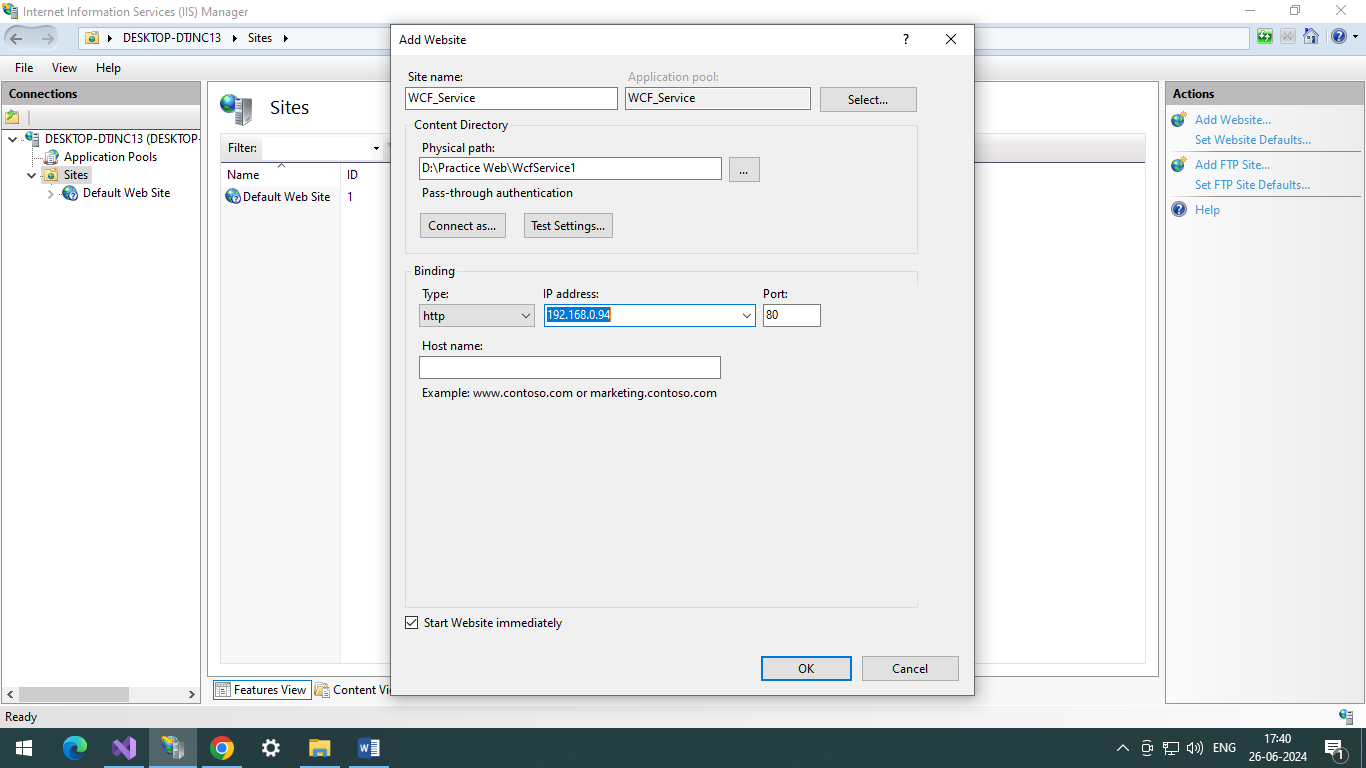
https://learn.microsoft.com/en-us/iis/application-frameworks/scenario-build-an-aspnet-website-on-iis/configuring-step-1-install-iis-and-asp-net-modules

1. Open IIS server, press Win+R command on keyboard and type inetmgr command. This will open IIS server application.

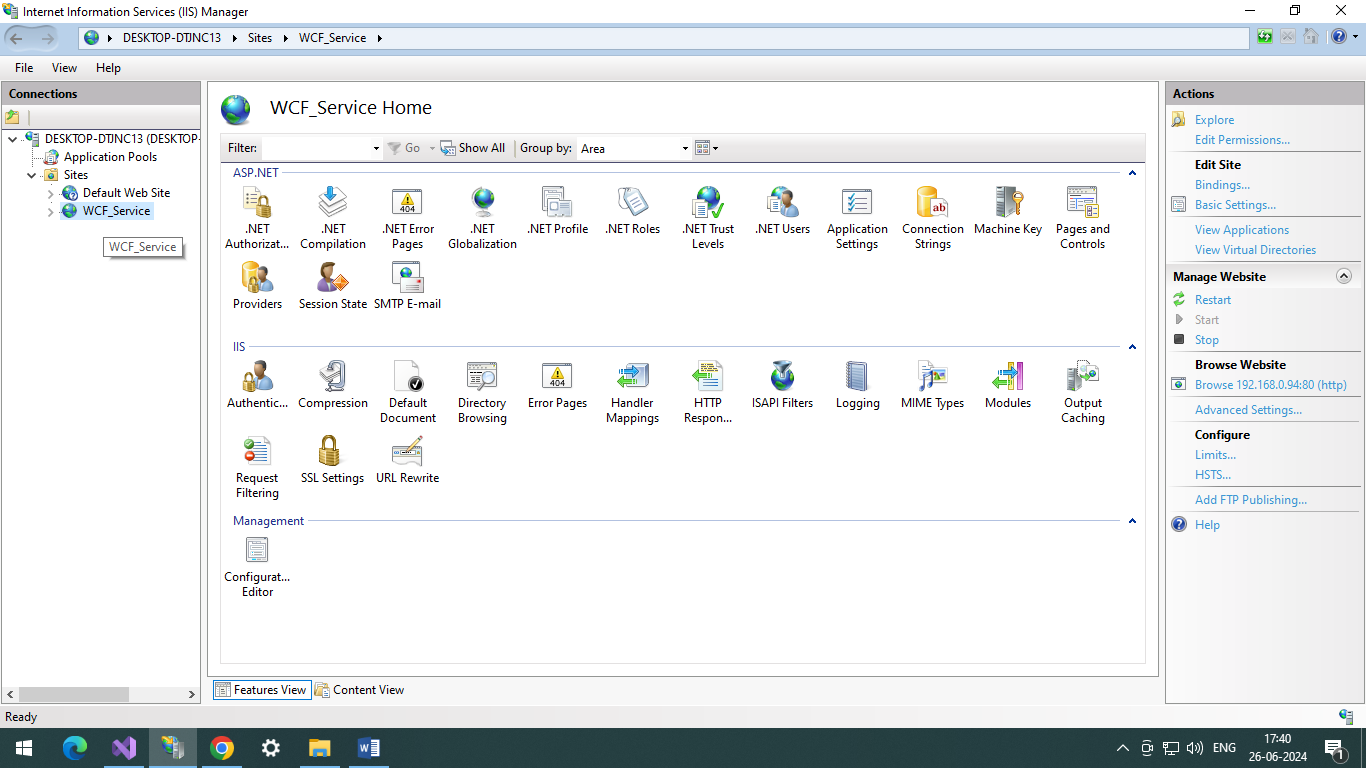




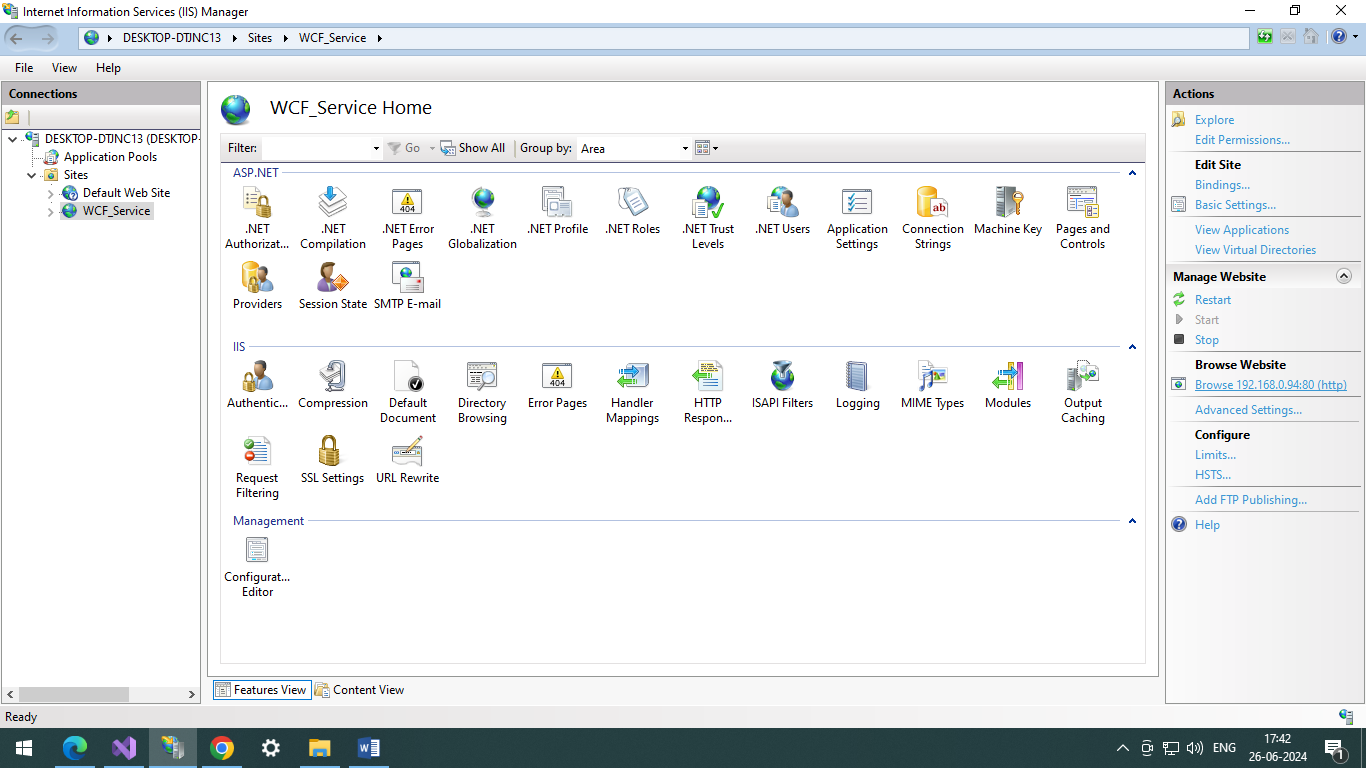
1. Now we have to create Web Application in IIS. Right click on Sites menu and click Add Website. Fill all the details required and click OK button.



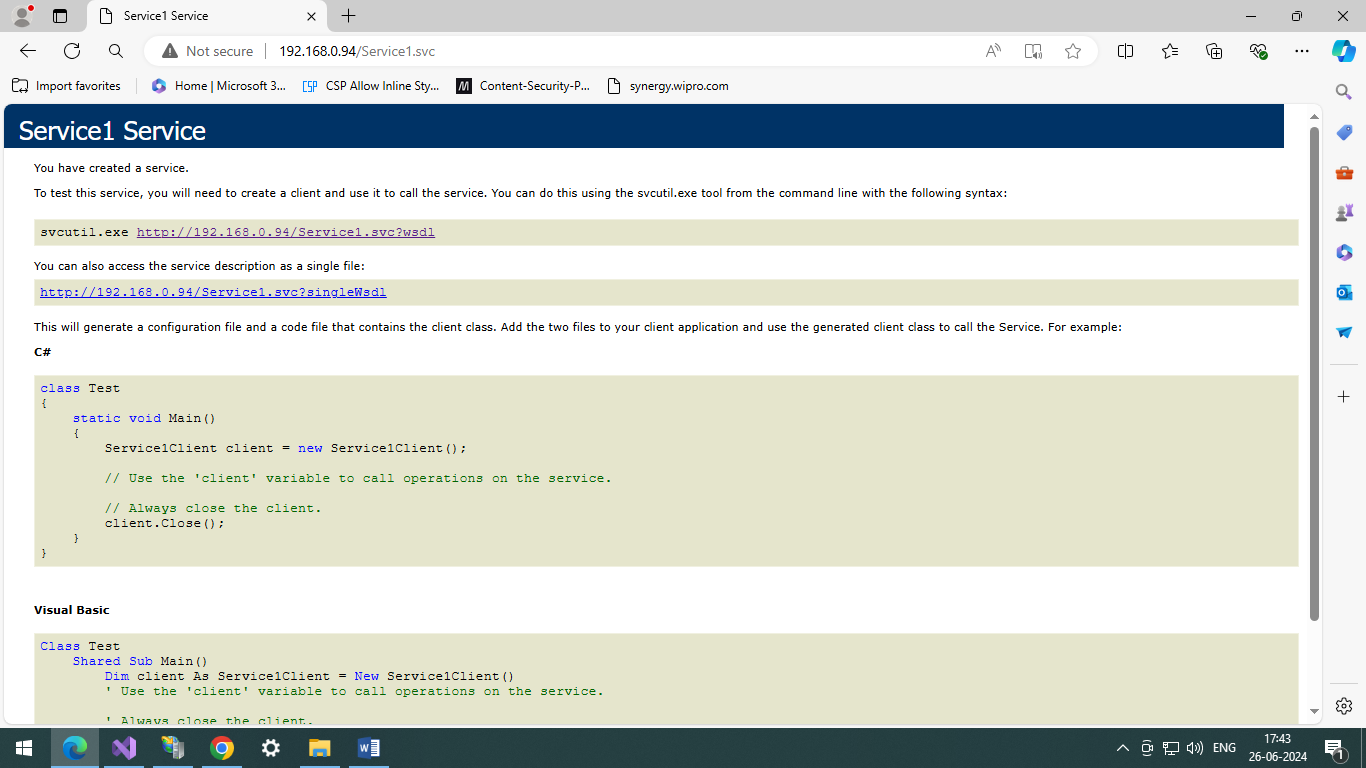
Wcf Service application will be added in the sites menu.



You can run the web application by clicking on the link below the Browse Website name.



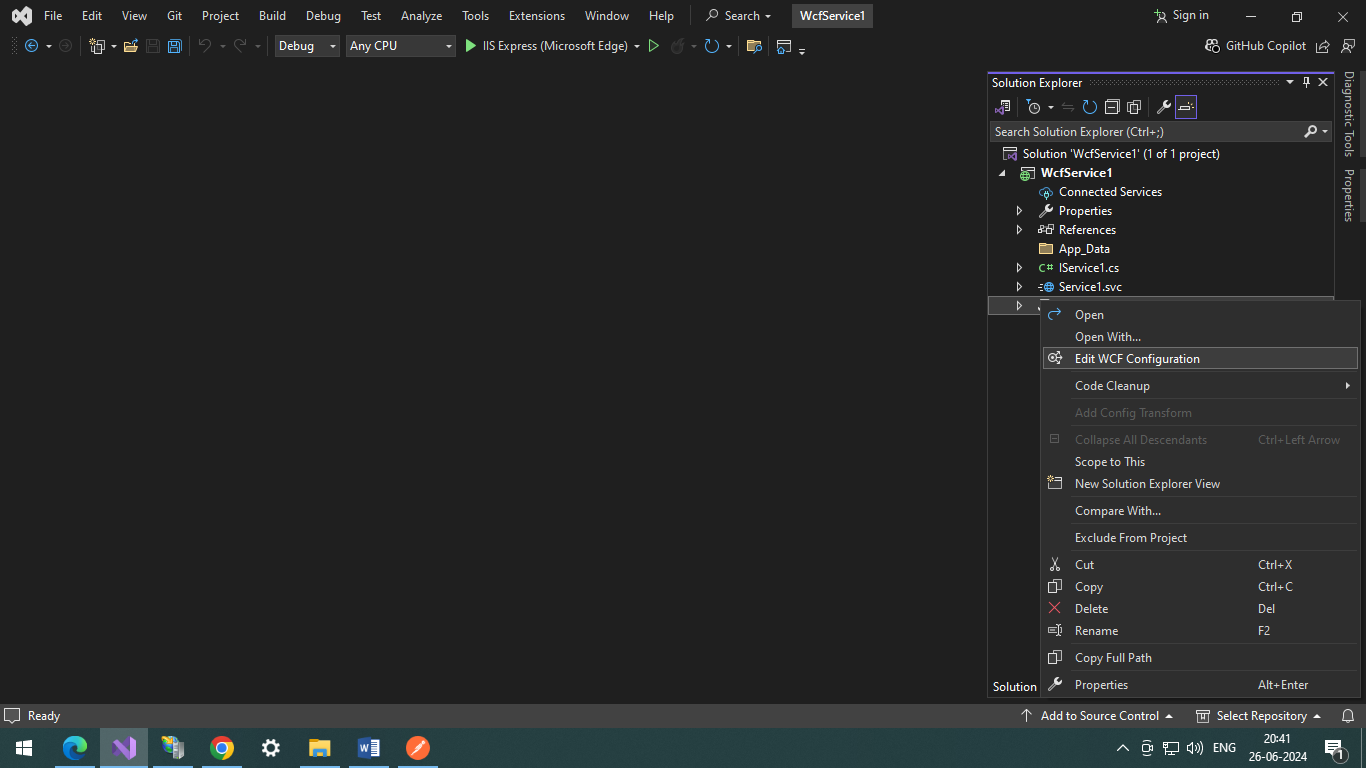
Your Wcf Application will look like this.

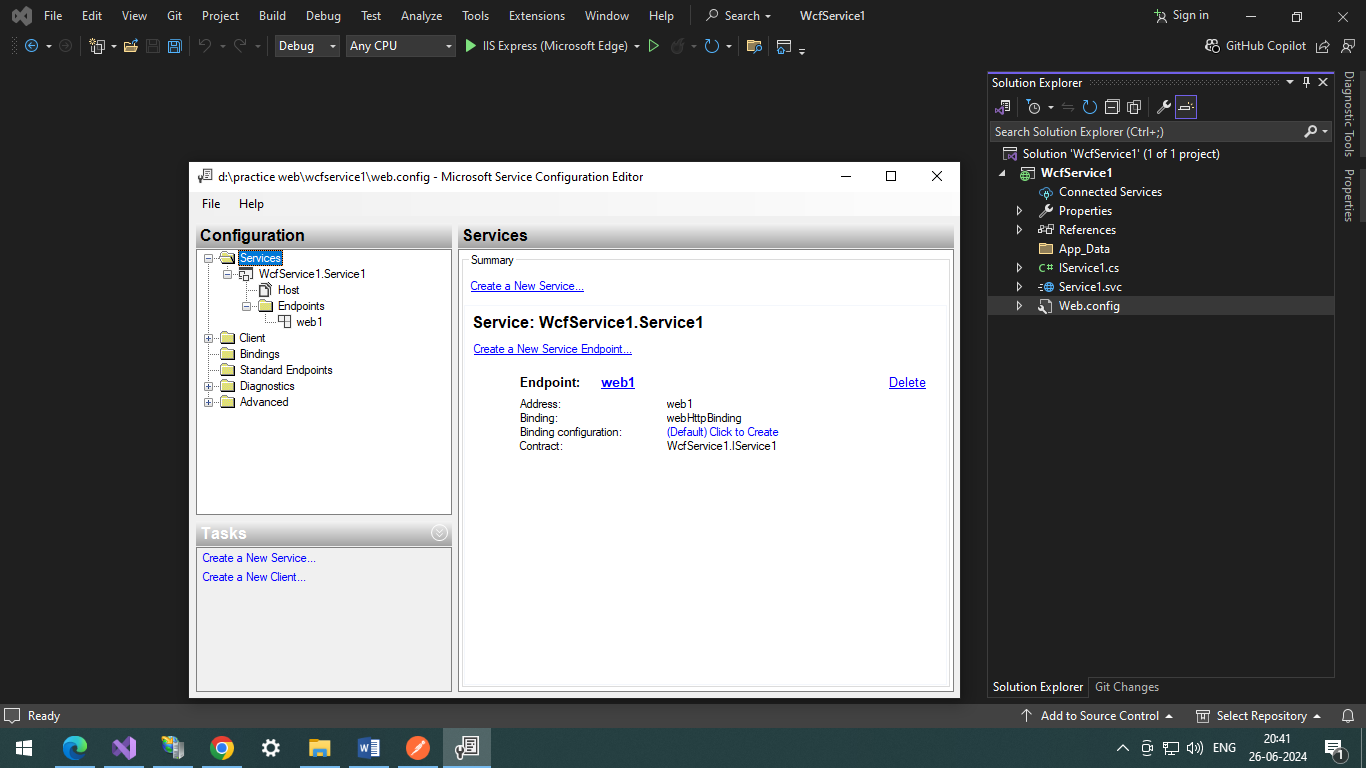


Configuring End Points in WCF Service.

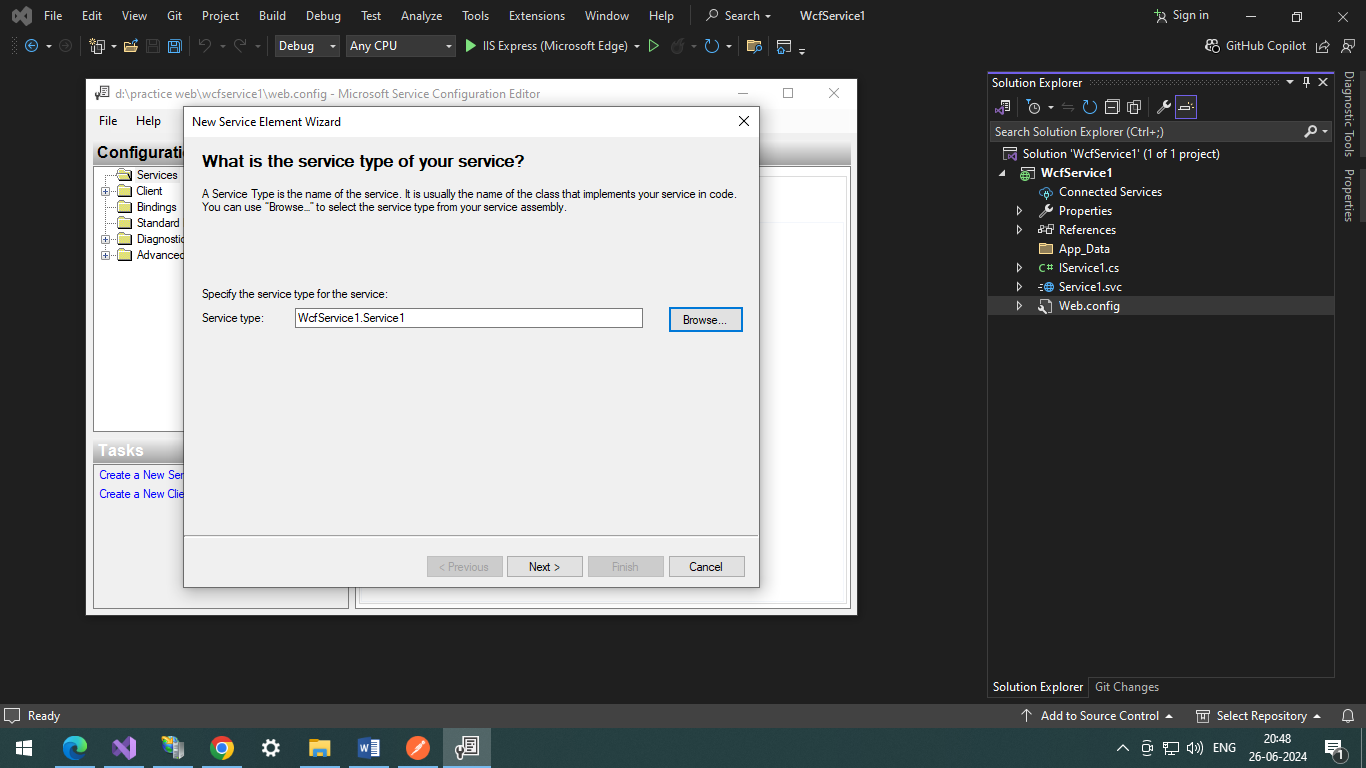
<https://www.c-sharpcorner.com/UploadFile/0c1bb2/endpoints-in-wcf/>

1. Edit Web.config

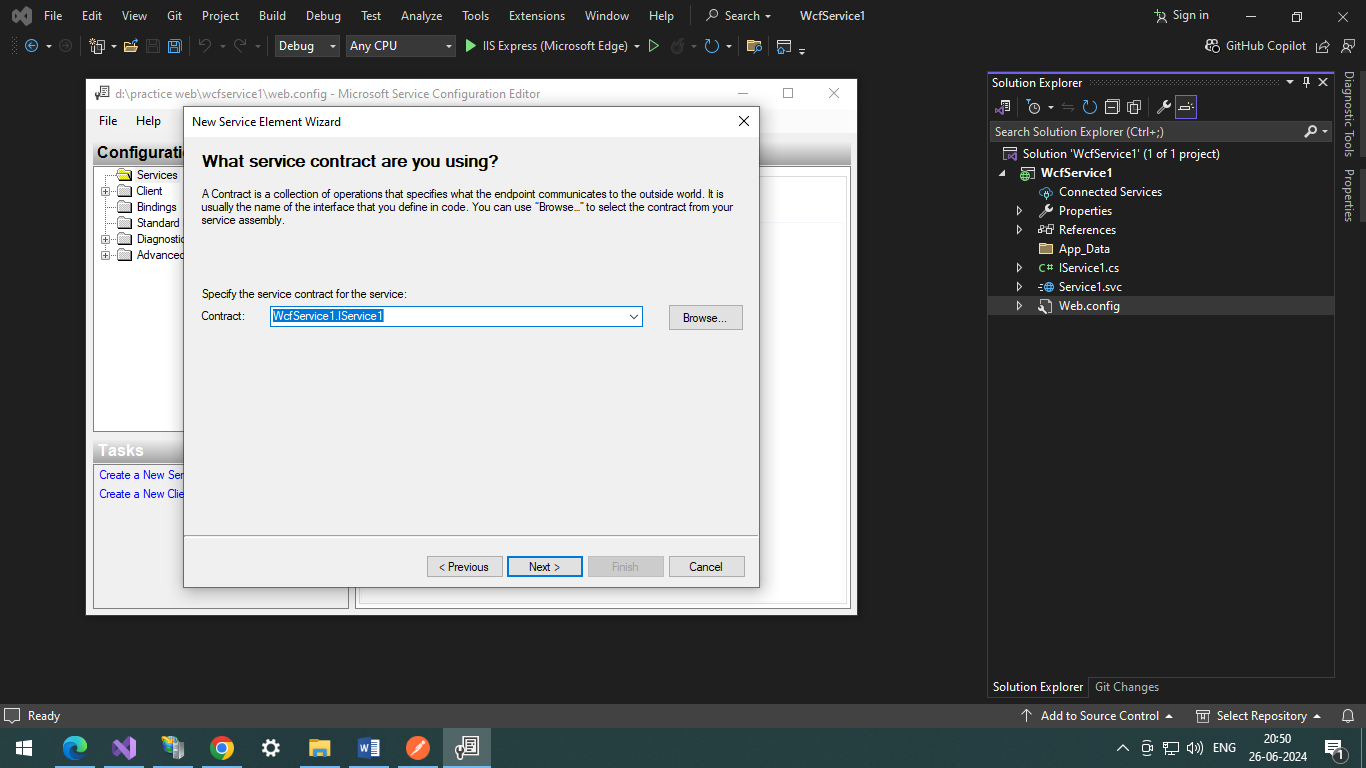




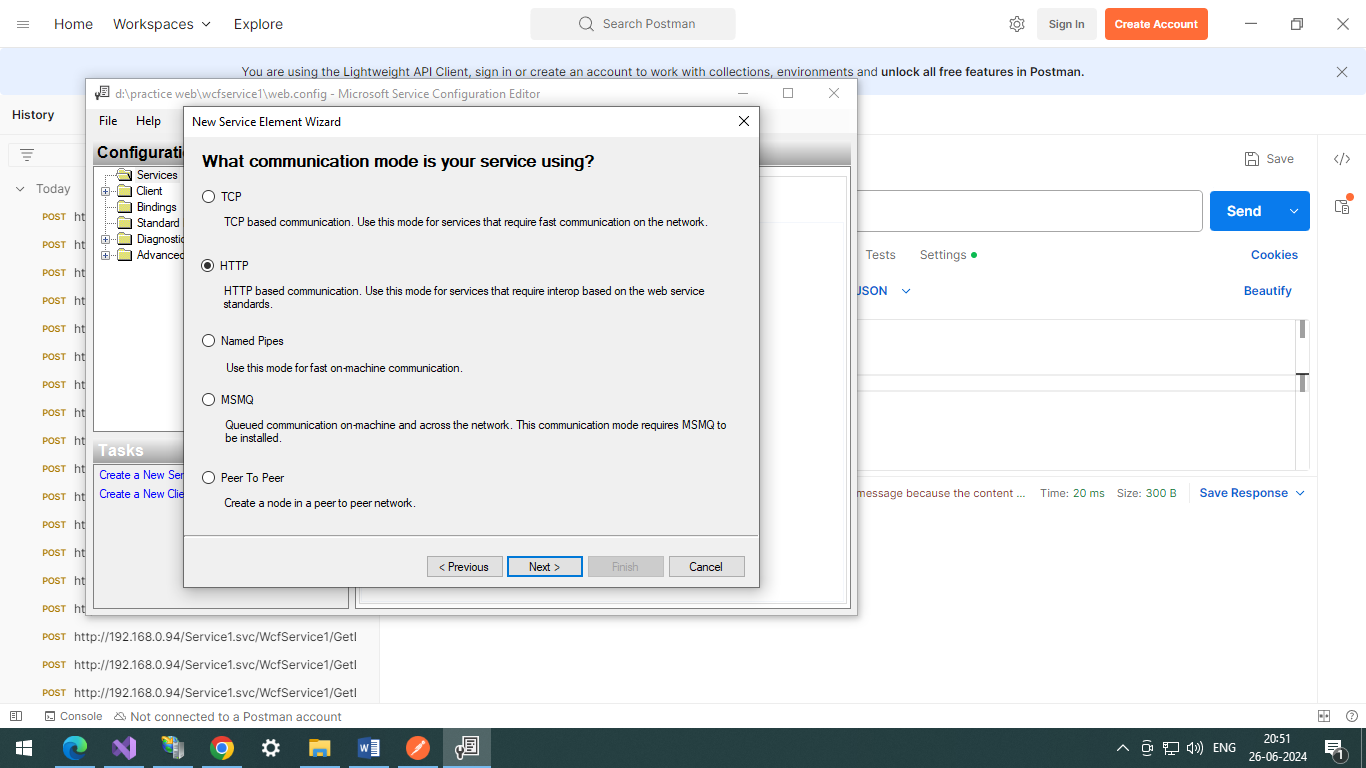
1. Click on Create New Service  
   Specify the Service type for the service name by clicking Browse button and selecting the dll file.



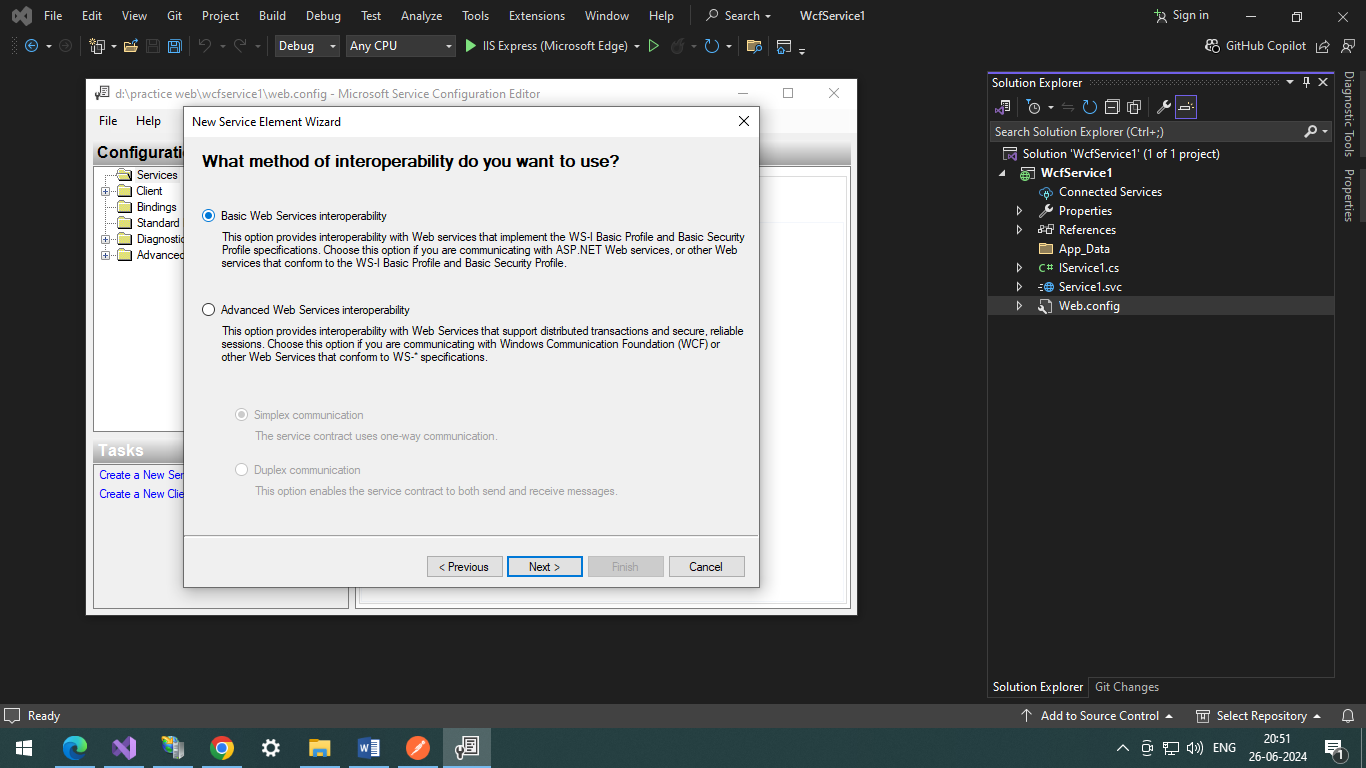
Specify the service contract for the service, it will come automatically.



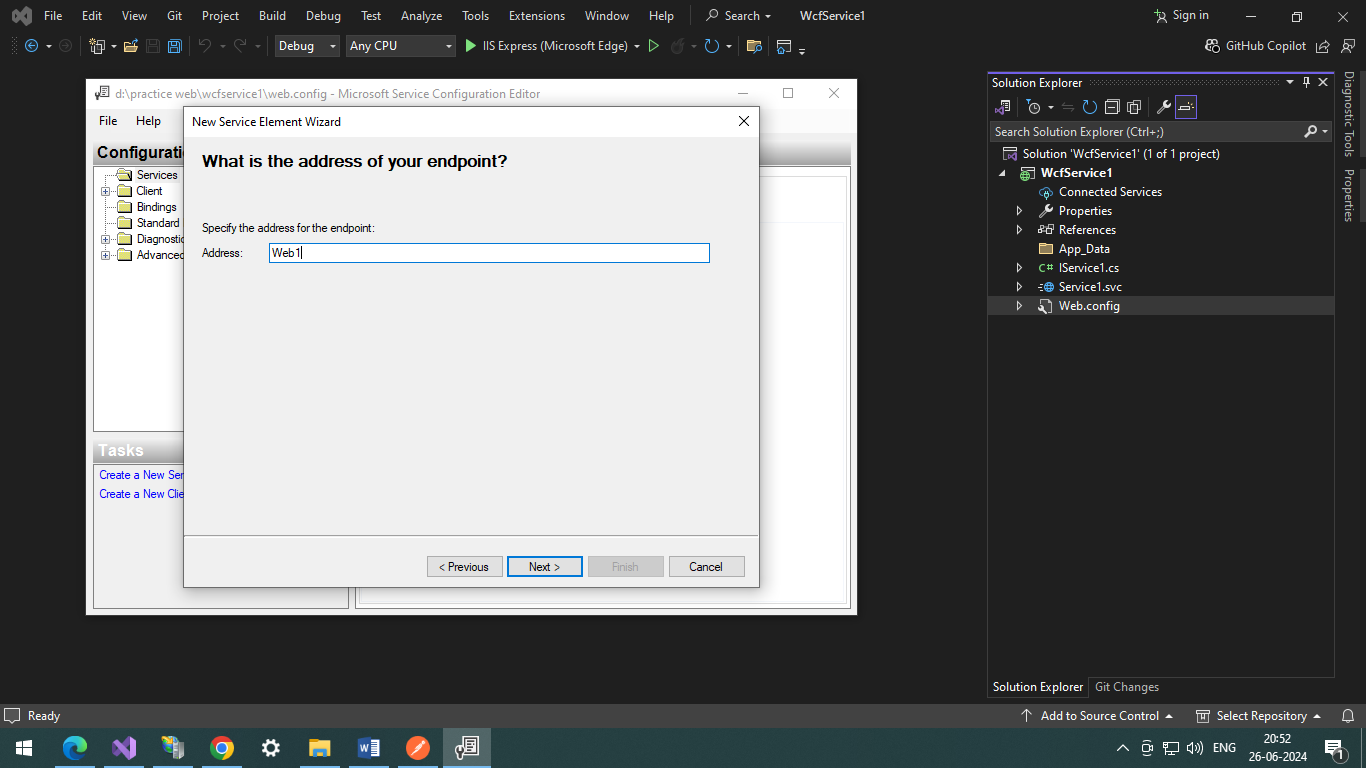
Select HTTP



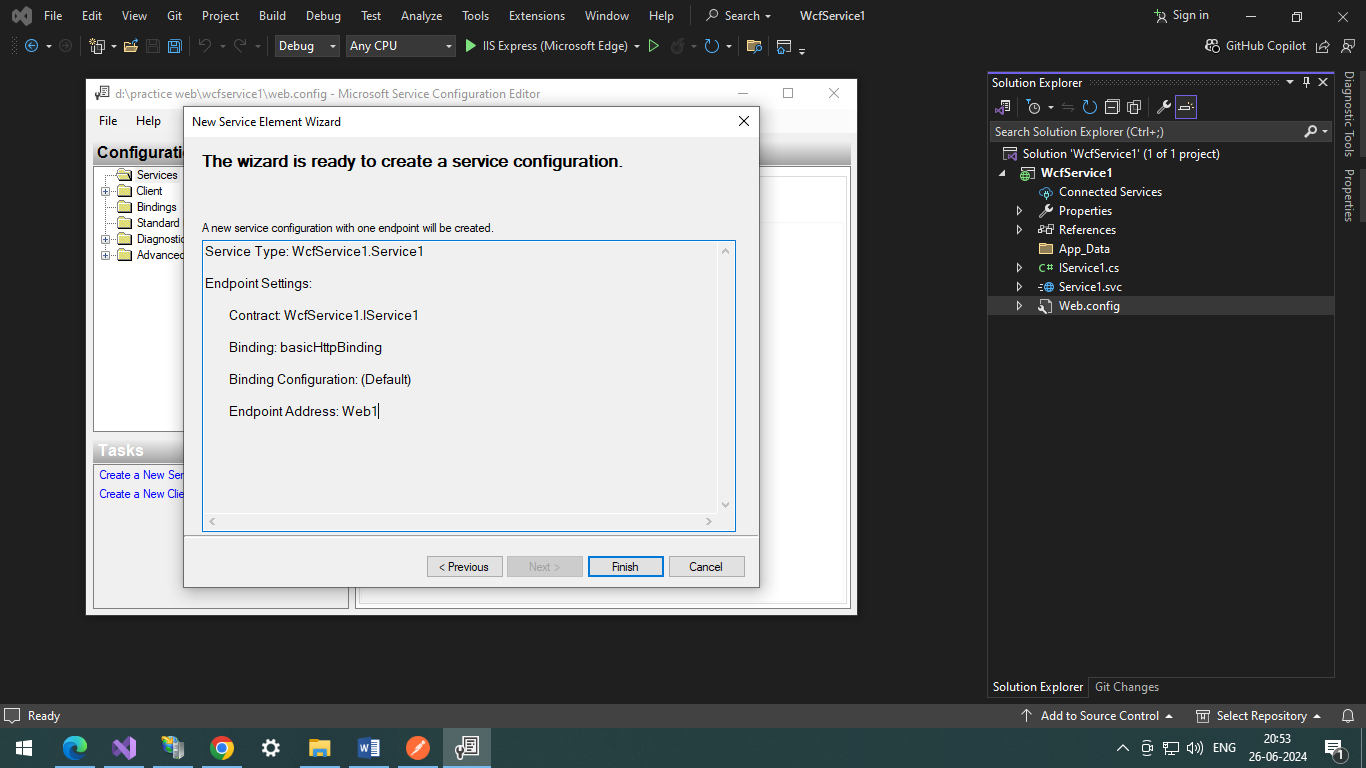
Select Basic Web Services interoperability.



Specify the relative address name

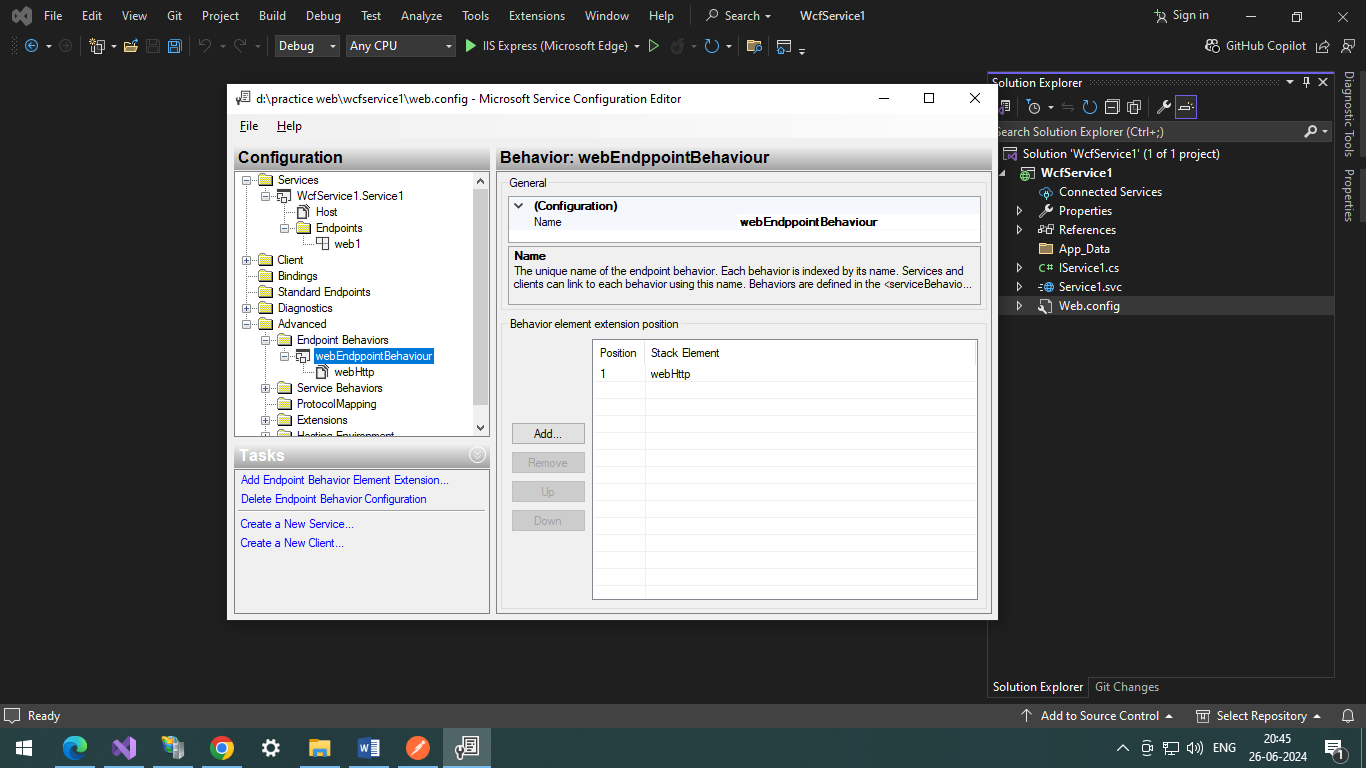


Finalize setting



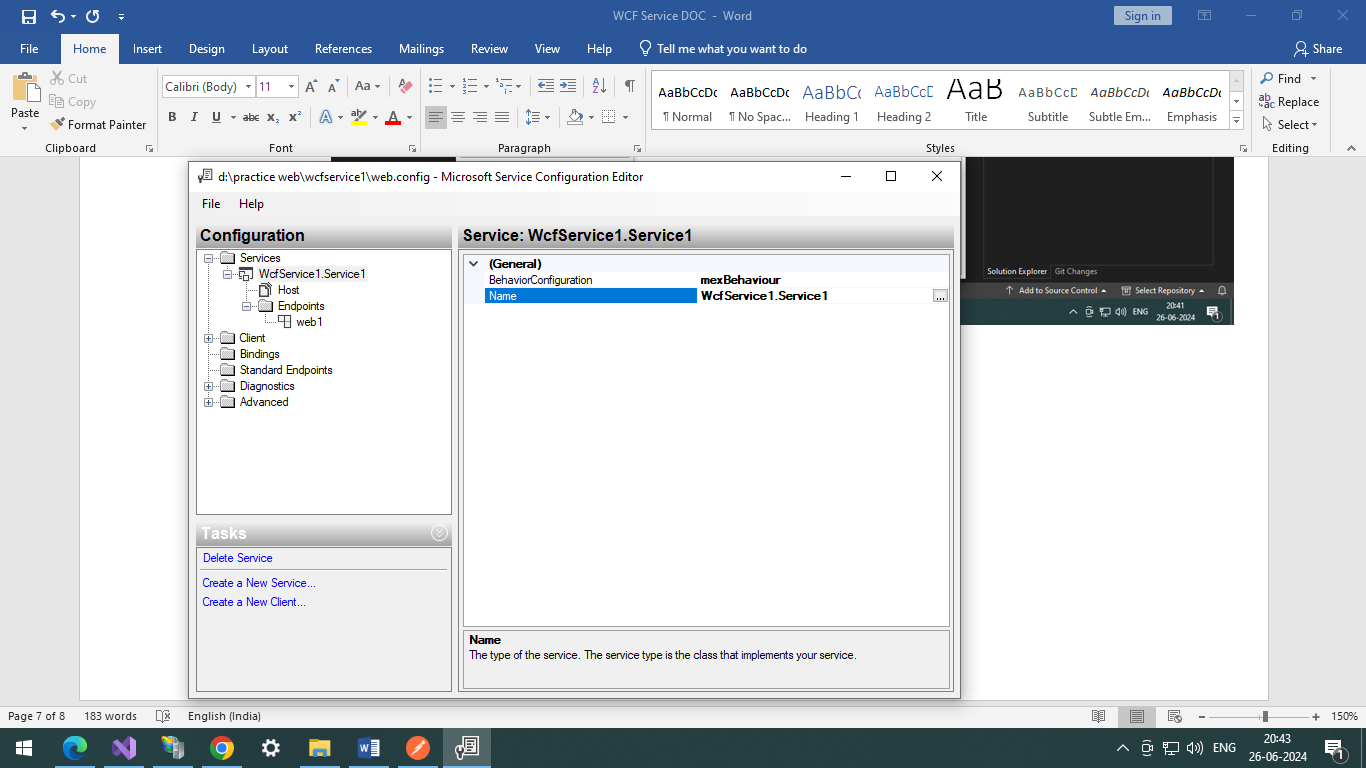
Change the basicHttpBinding to WebHttpBinding .

Add EndPoint Behaviour as webHttp in Advance Menu

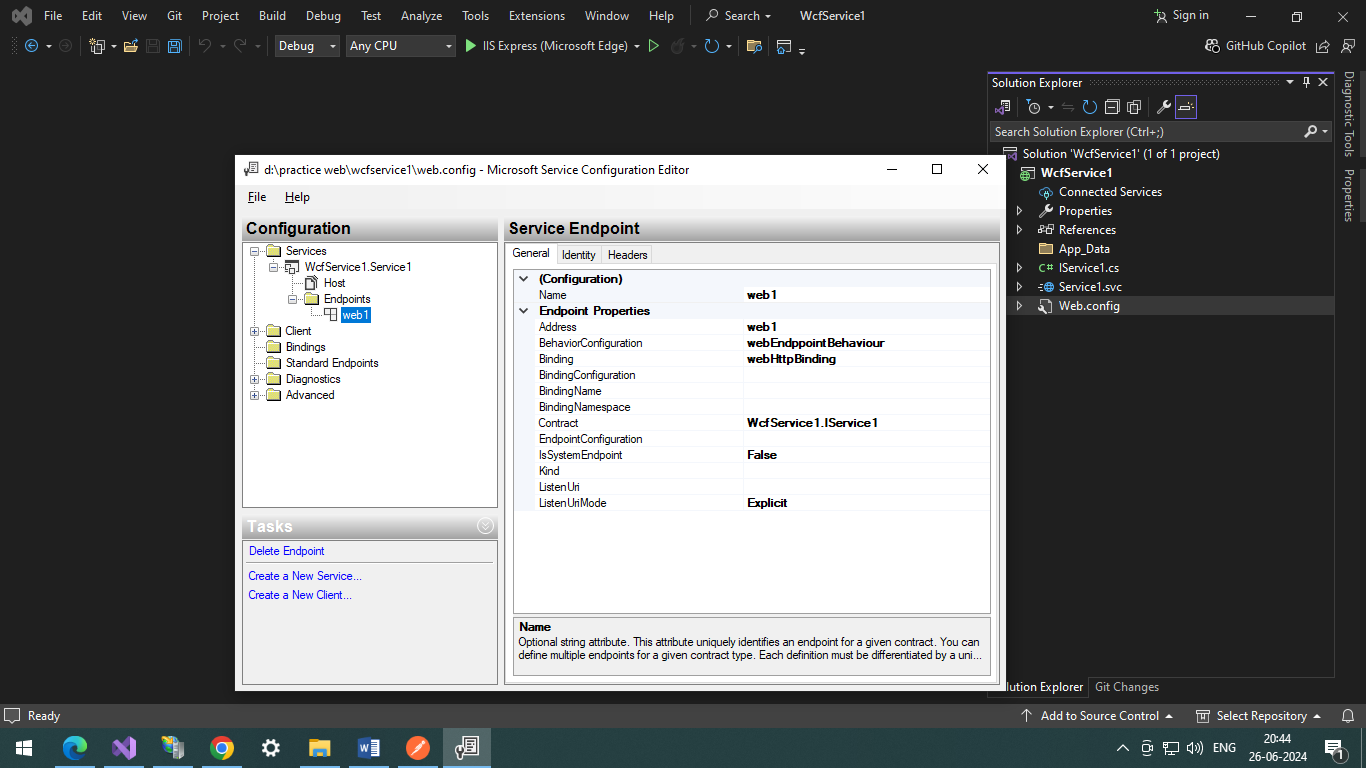


After finalizing , setting should look similar to below images.

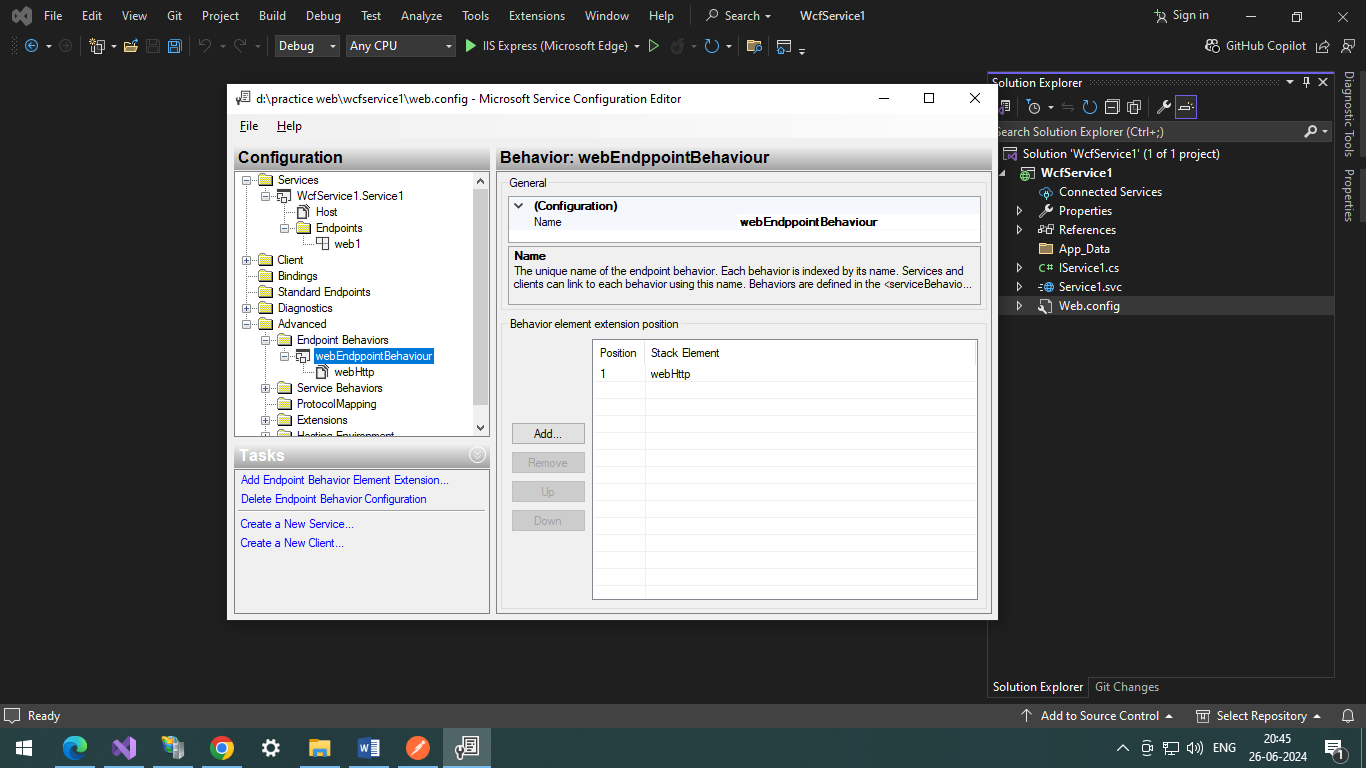
Specify the Service Name



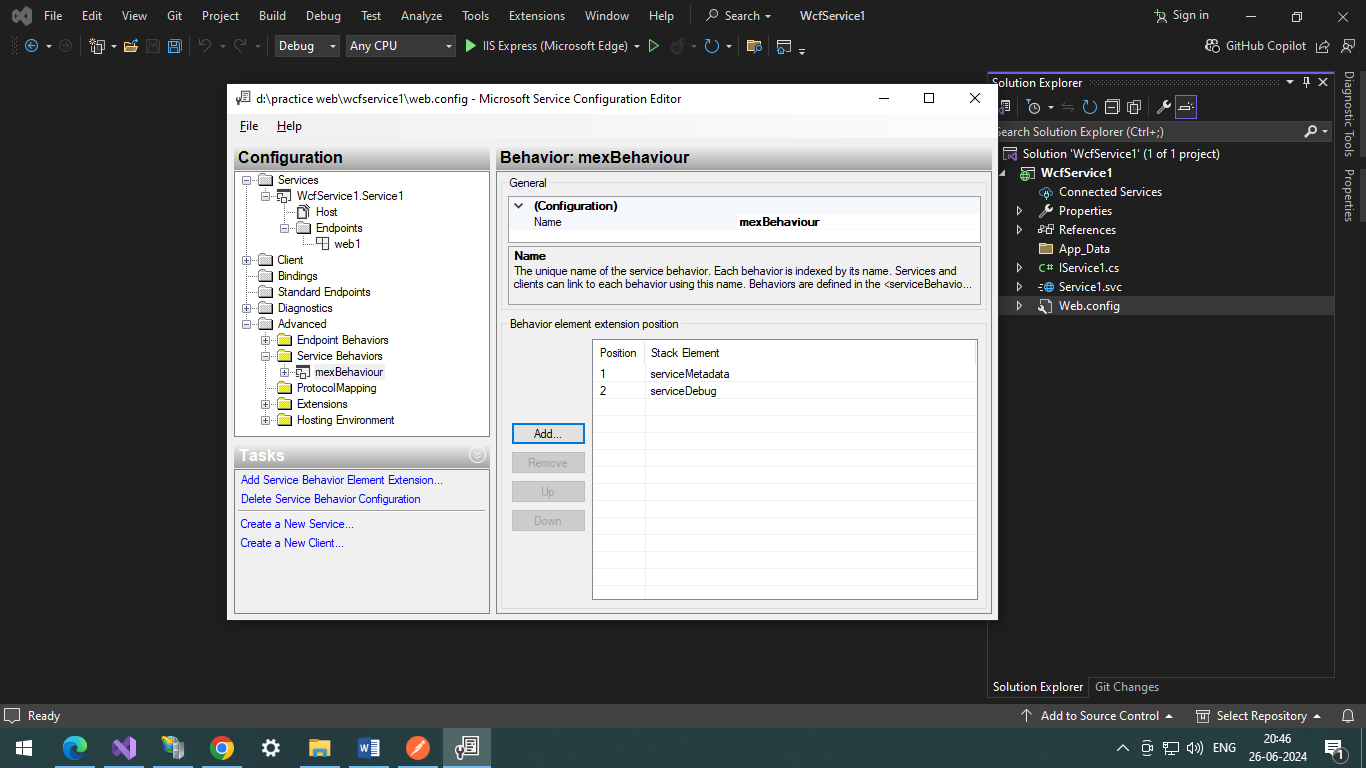
Add New End Point



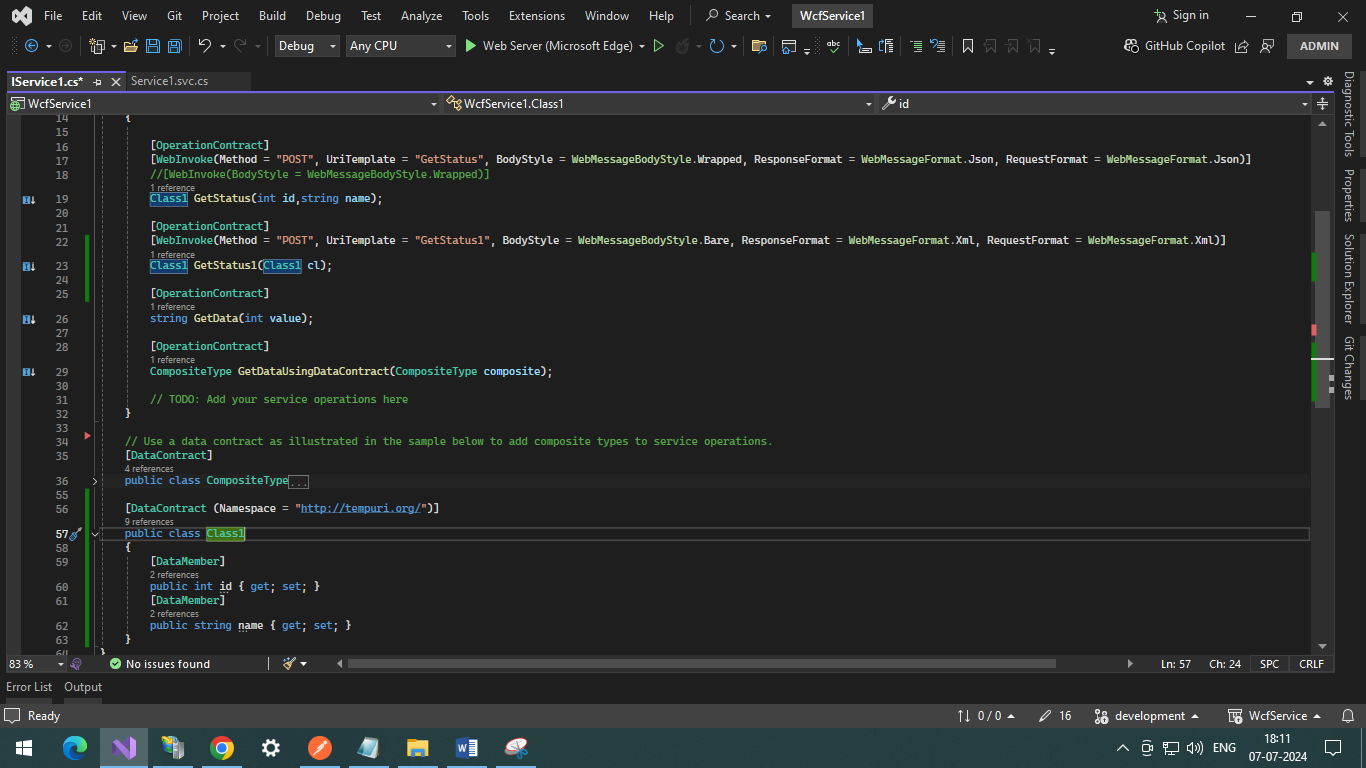
Add End Point Behaviour as webHttp in Advance Menu



Add or Specify Service Behaviour name in Advance Menu

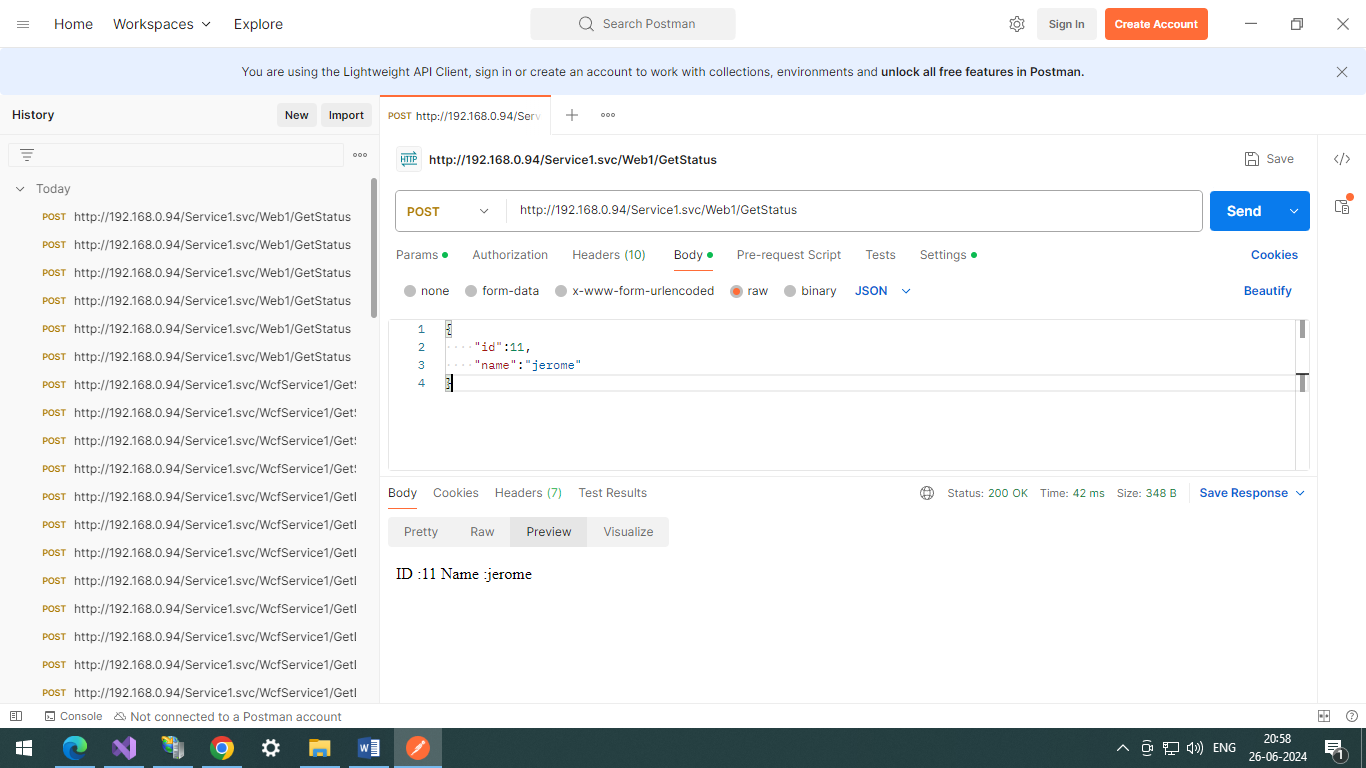


CONFIGURATION FOR JSON AND XML REQUEST RESPONSE

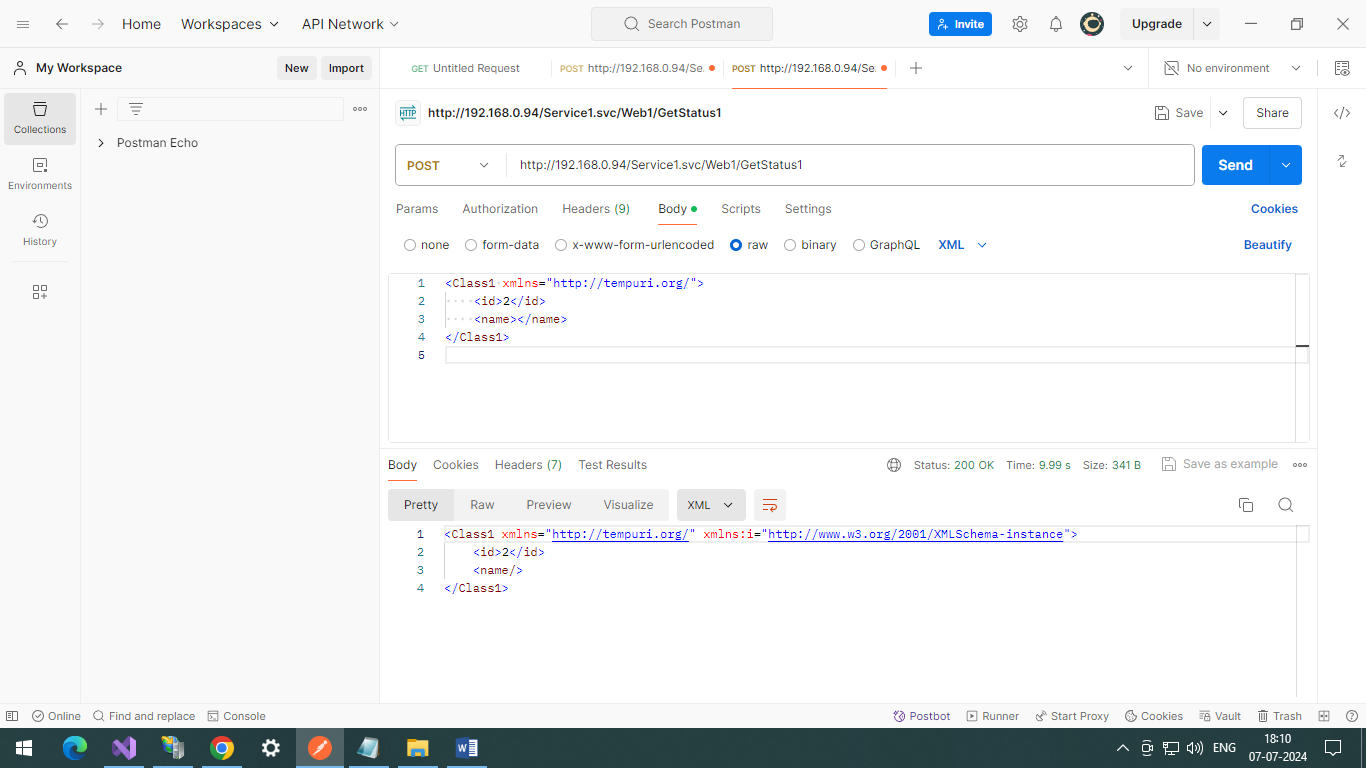


Output in Postman

For JSON values



For XML REQUEST RESPONSE



Consume XML Service Request in C# console application.

internal class Program

{

static void Main(string[] args)

{

int id = 0;

string name = "";

Console.WriteLine("Enter ID :");

id = Convert.ToInt32(Console.ReadLine());

Console.WriteLine("Enter Name");

name = Console.ReadLine();

ConsumeWCF\_XML\_Service(id,name);

Console.ReadLine();

}

public static async void ConsumeWCF\_XML\_Service(int id,string name)

{

try

{

var client = new HttpClient();

var request = new HttpRequestMessage(HttpMethod.Post, "http://192.168.0.94/Service1.svc/Web1/GetStatus1");

var content = new StringContent("<Class1 xmlns=\"http://tempuri.org/\"><id>"+id+"</id><name>"+name+"</name></Class1>", null, "application/xml");

request.Content = content;

var response = await client.SendAsync(request);

response.EnsureSuccessStatusCode();

Console.WriteLine(await response.Content.ReadAsStringAsync());

}

catch (Exception ex)

{

Console.WriteLine(ex.ToString());

}

}

}