

How to make Web API using ASP.NET Core 7.0 in Visual Studio 2022 Community Edition

This tutorial will guide you through the process of making a RESTful Web API using ASP.net Core 7.0 using Visual Studio 2022 Community Edition

Web API will have a GET Request GetContacts which will return a list of Contacts.

Visual Studio 2022 uses a software named as Swagger to test Web API. It is just like POSTMAN.

First of all create a new ASP.Net Core Web API Project in Visual Studio 2022 Community Edition.

Start Visual Studio 2022 Community Edition and Click on Create a New Project.



Select the above option and Click on Next

Configure your new project

ASP.NET Core Web API C# Linux macOS Windows Cloud Service Web WebAPI

Project name

WebApplication4

Location

C:\Users\raman\source\repos

Solution name ⓘ

WebApplication4

☐ Place solution and project in the same directory

Project will be created in "C:\Users\raman\source\repos\WebApplication4\WebApplication4\"

Click on Next

Additional information

ASP.NET Core Web API

C#

Linux

macOS

Windows

Cloud

Service

Web

WebAPI

Framework ⓘ

.NET 7.0 (Standard Term Support)

Authentication type ⓘ

None

☒ Configure for HTTPS ⓘ

☐ Enable Docker ⓘ

Docker OS ⓘ

Linux

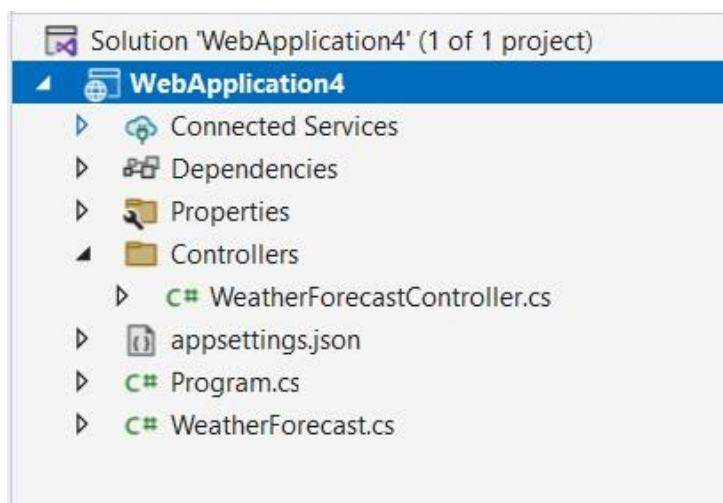
☒ Use controllers (uncheck to use minimal APIs) ⓘ

☒ Enable OpenAPI support ⓘ

☐ Do not use top-level statements ⓘ

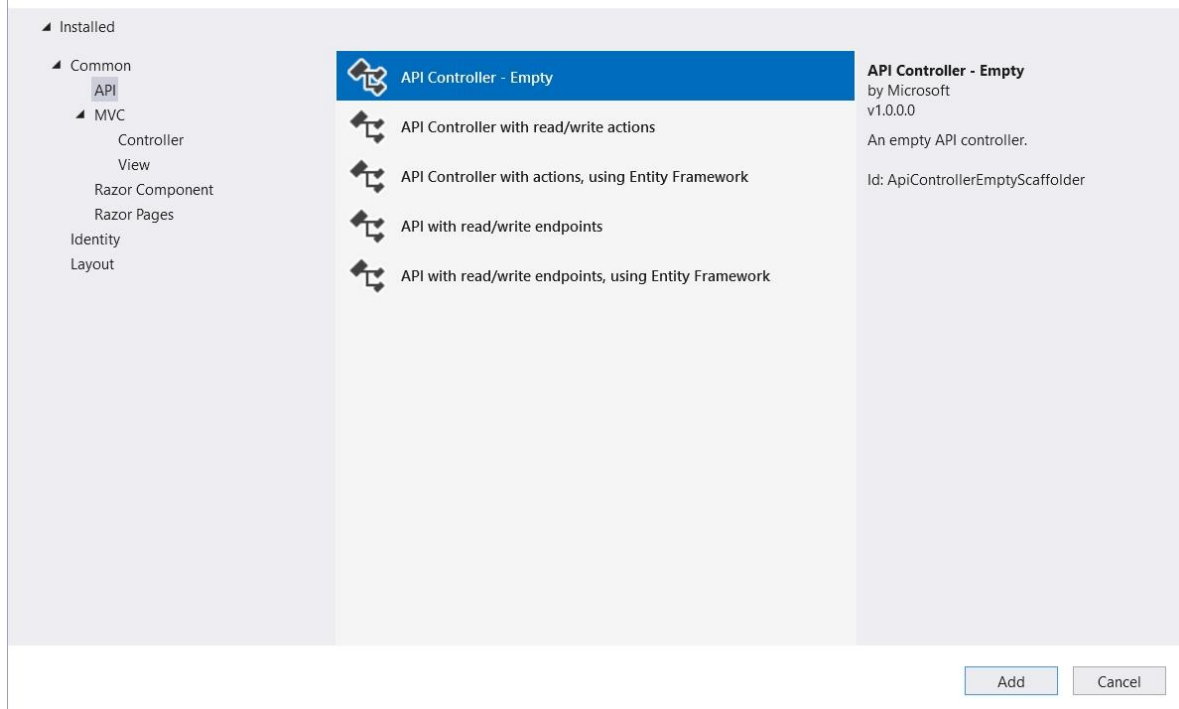
Click on Create and a New Project will be created

And Project Structure will look like

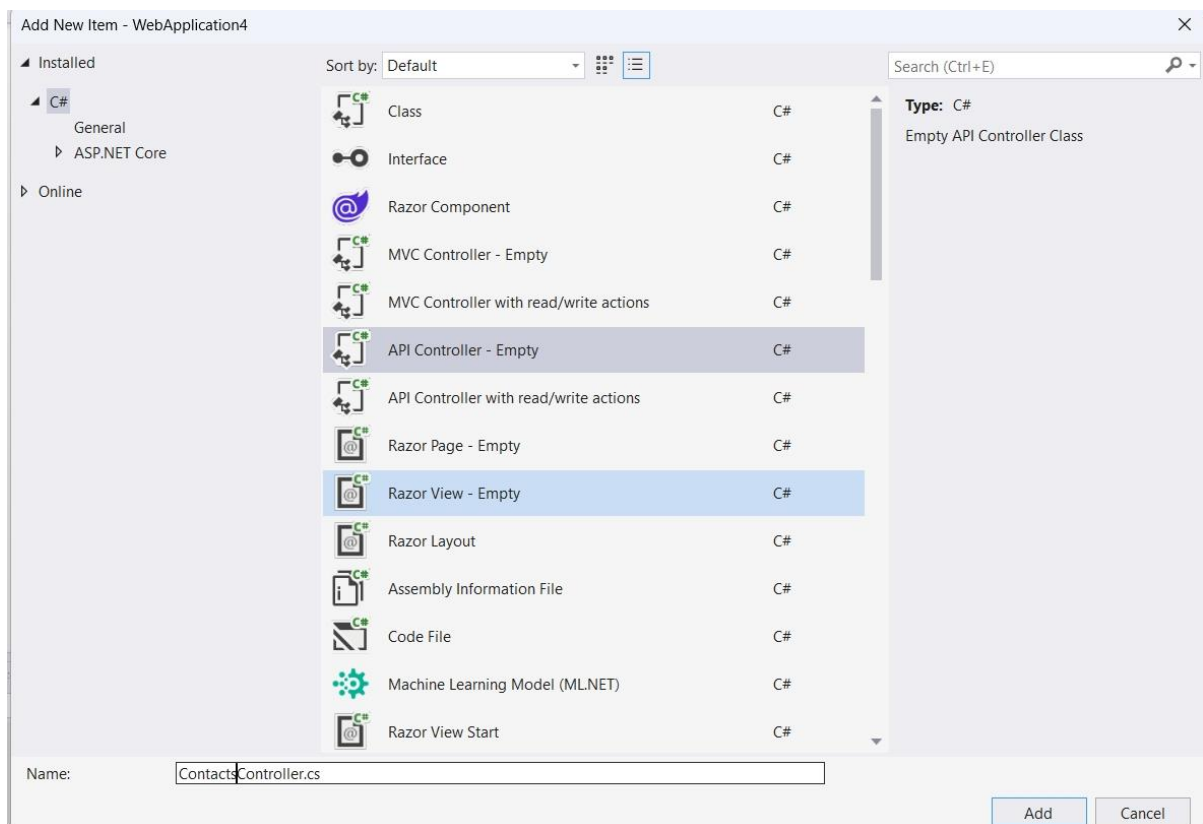


Add a New Controller to Controllers as shown below

Add New Scaffolded Item



Click on Add



Following is the code for ContactsController.cs

```
using Microsoft.AspNetCore.Http;  
using Microsoft.AspNetCore.Mvc;
```

```

using WebApplication4.Models;

namespace WebApplication4.Controllers
{
    [Route("api/GetContacts")]
    [ApiController]
    public class ContactsController : ControllerBase
    {
        [HttpGet]
        public IEnumerable<Contact> GetContacts()
        {
            return new List<Contact>
            {
                new Contact { Id = 1, Name="Raman", Address="ábcdef
address", Email="raman@dfgh.com", Phone="898989"},
                new Contact { Id = 2, Name="Raman Deep", Address="ábcdef
address1", Email="ramandeep@dfgh.com", Phone="4567898989"},
                new Contact { Id = 3, Name="Raman Deep Singh", Address="ábcdefrds
address", Email="ramanrds@dfgh.com", Phone="123458989"}
            };
        }
    }
}

```

Add a new folder in the project with name Models

In folder Models create a new Class Contact.cs

Code for Contact.cs is

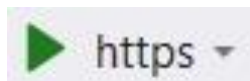
```

namespace WebApplication4.Models
{
    public class Contact
    {
        public int Id { get; set; }
        public string Name { get; set; }
        public string Address { get; set; }
        public string Email { get; set; }
        public string Phone { get; set; }
    }
}

```

Now how to run project or see output

Click on



Now after successful run swagger will open up like this

WebApplication4 ^{1.0} OAS3

<https://localhost:7101/swagger/v1/swagger.json>

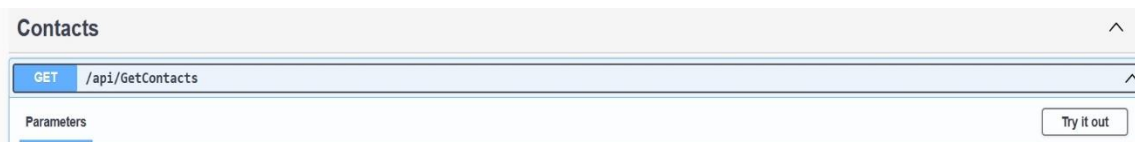
Contacts

GET /api/GetContacts

WeatherForecast

GET /WeatherForecast

Now click on GET Button next to /api/GetContacts



Click on Try It Out Button and then Click on Execute

You will see the following output

Server response

Code

Details

200

Response body

```
[
  {
    "id": 1,
    "name": "Raman",
    "address": "ábcdef address",
    "email": "raman@dfgh.com",
    "phone": "898989"
  },
  {
    "id": 2,
    "name": "Raman Deep",
    "address": "ábcdef address1",
    "email": "ramandeep@dfgh.com",
    "phone": "4567898989"
  },
  {
    "id": 3,
    "name": "Raman Deep Singh",
    "address": "ábcdefrds address",
    "email": "ramanrds@dfgh.com",
    "phone": "123458989"
  }
]
```