**1.First Web Api using .Net core**

Create a .Net core web application with API template. Use the option to create controller with Read Write permissions. Notice the ValuesController creation with Action methods corresponding to the Action verbs.

On creation of the Web API, execute the application and check if the GET action method result is returned as expected.

a)Program.cs:

using Microsoft.OpenApi.Models;

var builder = WebApplication.CreateBuilder(args);

// Register services

builder.Services.AddControllers(); // 👈 Enables controller support

builder.Services.AddEndpointsApiExplorer();

builder.Services.AddSwaggerGen(c =>

{

    c.SwaggerDoc("v1", new OpenApiInfo { Title = "FirstWebApi", Version = "v1" });

});

var app = builder.Build();

if (app.Environment.IsDevelopment())

{

    app.UseSwagger();

    app.UseSwaggerUI();

}

// Optional: Disable HTTPS if it gives redirect issues

// app.UseHttpsRedirection();

app.UseAuthorization();

// 👇 Very important: enables controller routing

app.MapControllers();

app.Run();

b)ValuesController.cs:

using Microsoft.AspNetCore.Mvc;

using System.Collections.Generic;

namespace FirstWebApi.Controllers

{

    [ApiController]

    [Route("api/[controller]")]

    public class ValuesController : ControllerBase

    {

        [HttpGet]

        public IEnumerable<string> Get()

        {

            return new string[] { "value1", "value2" };

        }

        [HttpGet("{id}")]

        public string Get(int id)

        {

            return $"value {id}";

        }

        [HttpPost]

        public void Post([FromBody] string value)

        {

        }

        [HttpPut("{id}")]

        public void Put(int id, [FromBody] string value)

        {

        }

        [HttpDelete("{id}")]

        public void Delete(int id)

        {

        }

    }

}

c)appsettings.json:

{

  "Logging": {

    "LogLevel": {

      "Default": "Information",

      "Microsoft.AspNetCore": "Warning"

    }

  },

  "AllowedHosts": "\*"

}

d)appsettings.Development.json:

{

  "Logging": {

    "LogLevel": {

      "Default": "Information",

      "Microsoft.AspNetCore": "Warning"

    }

  }

}

Output:



