**WebAPI\_HandsOn - 1**

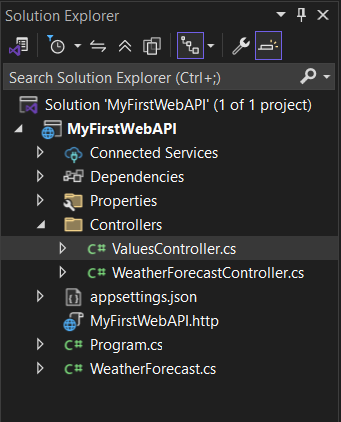
**First Web Api using .Net core**

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.

**Solution:-**

File Structure:



**Code:-**

using Microsoft.AspNetCore.Mvc;

using System.Collections.Generic;

namespace MyFirstWebAPI.Controllers

{

[ApiController]

[Route("api/[controller]")]

public class ValuesController : ControllerBase

{

private static List<string> values = new List<string> { "Apple", "Banana", "Orange"};

// GET: api/values

[HttpGet]

public ActionResult<IEnumerable<string>> Get()

{

return Ok(values);

}

// GET: api/values/1

[HttpGet("{id}")]

public ActionResult<string> Get(int id)

{

if (id < 0 || id >= values.Count)

return NotFound();

return Ok(values[id]);

}

// POST: api/values

[HttpPost]

public IActionResult Post([FromBody] string value){

values.Add(value);

return CreatedAtAction(nameof(Get), new { id = values.Count - 1 }, value);

}

// PUT: api/values/1

[HttpPut("{id}")]

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

{

if (id < 0 || id >= values.Count)

return NotFound();

values[id] = value;

return NoContent();

}

// DELETE: api/values/1

[HttpDelete("{id}")]

public IActionResult Delete(int id)

{

if (id < 0 || id >= values.Count)

return NotFound();

values.RemoveAt(id);

return NoContent();

}

}

}

