4.Update Employee data as per the input thru Web API PUT action method call Employee information has to be updated based on the user input. Use Swagger tool to invoke the action method mapped with Http PUT action verb to update an employee data. Modify the action method to return Employee data thru ActionResult. Check if the id value is lesser than or equal to 0. If true, throw BadRequest action result with the message ‘Invalid employee id’ If the value is greater than 0 but not available in the list of employee ids that is there in the hardcoded list of employees, throw BadRequest action result with the same message as stated above. If the id value is valid, use the JSON data from the input body and update the hardcoded list. Filter the employee list data for the input id and return that as the output.

CODE:

Empolyee.cs:

namespace FirstWebApi.Models

{

    public class Employee

    {

        public int Id { get; set; }

        public string? Name { get; set; }

        public string? Department { get; set; }

        public double Salary { get; set; }

    }

}

Empolyeecontroller.cs:  
using Microsoft.AspNetCore.Mvc;

using FirstWebApi.Models;

using System.Collections.Generic;

using System.Linq;

namespace FirstWebApi.Controllers

{

    [ApiController]

    [Route("api/[controller]")]

    public class EmployeesController : ControllerBase

    {

        // Hardcoded list

        private static List<Employee> employees = new List<Employee>

        {

            new Employee { Id = 1, Name = "John", Department = "HR", Salary = 50000 },

            new Employee { Id = 2, Name = "Sara", Department = "IT", Salary = 60000 },

            new Employee { Id = 3, Name = "David", Department = "Finance", Salary = 55000 }

        };

        // PUT api/employees/2

        [HttpPut("{id}")]

        public ActionResult<Employee> UpdateEmployee(int id, [FromBody] Employee updatedEmployee)

        {

            if (id <= 0)

            {

                return BadRequest("Invalid employee id");

            }

            var existingEmployee = employees.FirstOrDefault(e => e.Id == id);

            if (existingEmployee == null)

            {

                return BadRequest("Invalid employee id");

            }

            existingEmployee.Name = updatedEmployee.Name;

            existingEmployee.Department = updatedEmployee.Department;

            existingEmployee.Salary = updatedEmployee.Salary;

            return Ok(existingEmployee);

        }

    }

}

Program.cs:

using Microsoft.OpenApi.Models;

var builder = WebApplication.CreateBuilder(args);

builder.Services.AddControllers();

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(c =>

    {

        c.SwaggerEndpoint("/swagger/v1/swagger.json", "FirstWebApi v1");

    });

}

app.UseHttpsRedirection();

app.UseAuthorization();

app.MapControllers();

app.Run();

output:

Building...

info: Microsoft.Hosting.Lifetime[14]

Now listening on: http://localhost:5036

info: Microsoft.Hosting.Lifetime[0]

Application started. Press Ctrl+C to shut down.

info: Microsoft.Hosting.Lifetime[0]

Hosting environment: Development

info: Microsoft.Hosting.Lifetime[0]

Content root path: C:\Users\rahul\FirstWebApi\EmployeeApi





