**Boppudi Vivek(916243)**

1. Web Api CRUD operation

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=>**

**Employee.cs:**

**using System;**

**using System.Collections.Generic;**

**using System.Linq;**

**using System.Threading.Tasks;**

**namespace WebApiDay2Handson3.Models**

**{**

**public class Employee**

**{**

**public int Id { get; set; }**

**public string Name { get; set; }**

**}**

**}**

**EmployeeController.cs:**

**using Microsoft.AspNetCore.Mvc;**

**using Microsoft.AspNetCore.Http;**

**using Microsoft.AspNetCore.Authorization;**

**using System;**

**using System.Collections.Generic;**

**using System.Linq;**

**using System.Threading.Tasks;**

**using WebApiDay2Handson3.Models;**

**namespace WebApiDay2Handson3.Controllers**

**{**

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

**[ApiController]**

**public class EmployeesController : ControllerBase**

**{**

**private static List<Employee> \_emp = new List<Employee>();**

**// GET: api/<ValuesController1>**

**[HttpGet(Name = "GetAllStudent")]**

**public IActionResult Get()**

**{**

**return new ObjectResult(\_emp);**

**}**

**// GET api/<ValuesController1>/5**

**[HttpGet("{id}", Name = "GetStudent")]**

**public IActionResult Get(int id)**

**{**

**return new ObjectResult(\_emp.FirstOrDefault(p => p.Id == id));**

**}**

**// POST api/<ValuesController1>**

**[HttpPost(Name = "CreateStudent")]**

**public IActionResult Post([FromBody] Employee emps)**

**{**

**\_emp.Add(emps);**

**return CreatedAtRoute("GetStudent", new { id = emps.Id }, emps);**

**}**

**// PUT api/<ValuesController1>/5**

**[HttpPut("{id}", Name = "UpdateStudent")]**

**public IActionResult Put(int id, [FromBody] Employee emps)**

**{**

**\_emp.FirstOrDefault(p => p.Id == id).Name = emps.Name;**

**return CreatedAtRoute("GetStudent", new { id = emps.Id }, emps);**

**}**

**// DELETE api/<ValuesController1>/5**

**[HttpDelete("{id}", Name = "DeleteStudent")]**

**public IActionResult Delete(int id)**

**{**

**var \_emps = \_emp.FirstOrDefault(p => p.Id == id);**

**\_emp.Remove(\_emps);**

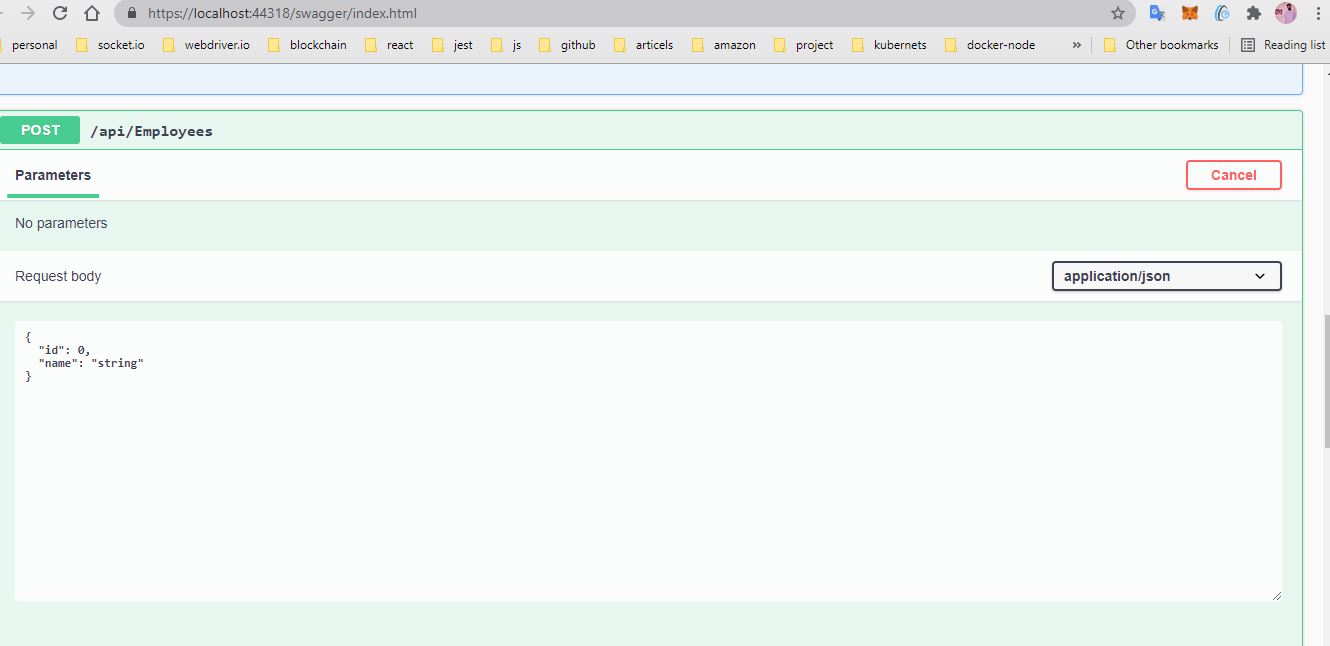
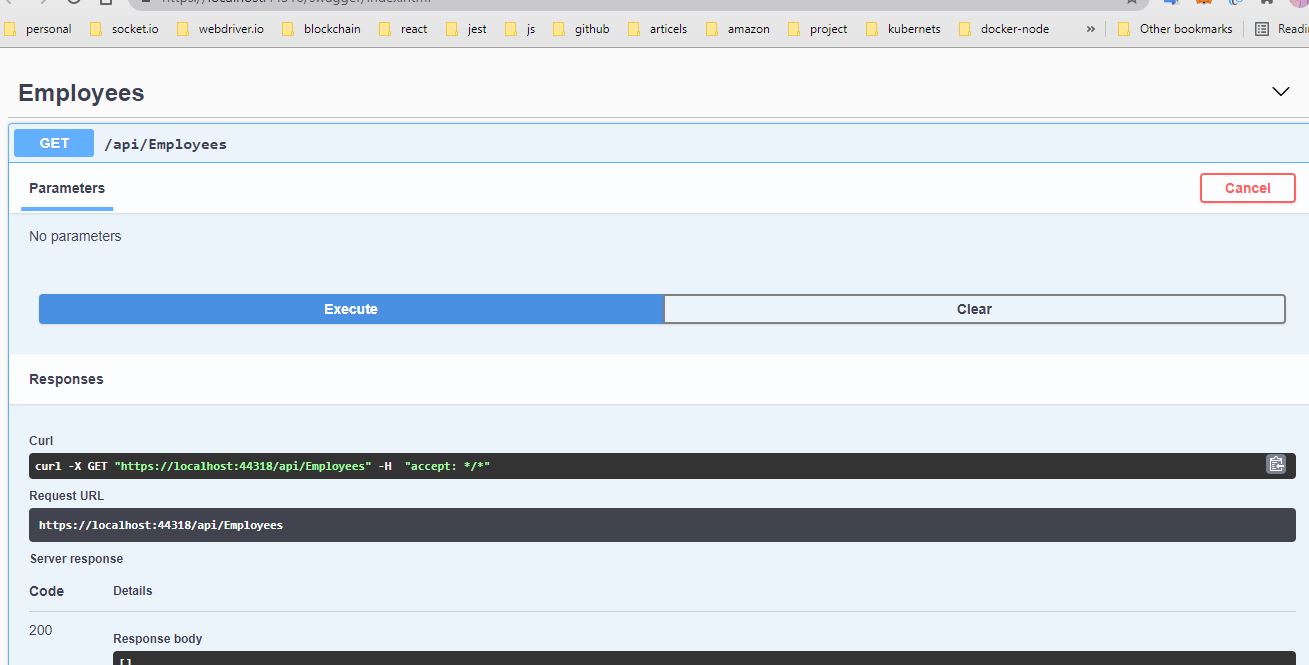
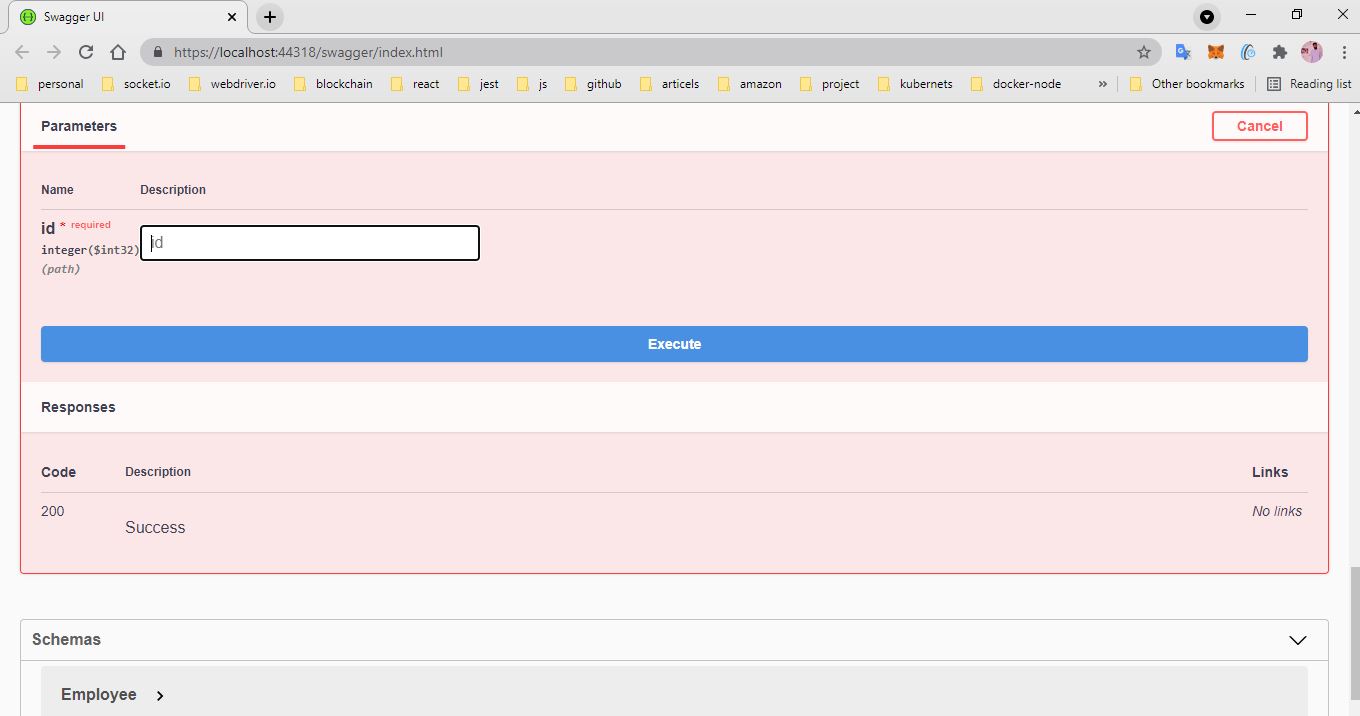
**return new NoContentResult();**

**}**

**}**

**}**

**Output:**

****