**3.WebApi\_Handson**

**Code:**

1)using System;

using System.Collections.Generic;

namespace SwaggerDemoApi.Models

{

public class Employee

{

public int Id { get; set; }

public string Name { get; set; }

public int Salary { get; set; }

public bool Permanent { get; set; }

public Department Department { get; set; }

public List<Skill> Skills { get; set; }

public DateTime DateOfBirth { get; set; }

}

}

Depart ment.cs

namespace SwaggerDemoApi.Models

{

public class Department

{

public int Id { get; set; }

public string Name { get; set; }

}

}

Skill.cs

namespace SwaggerDemoApi.Models

{

public class Skill

{

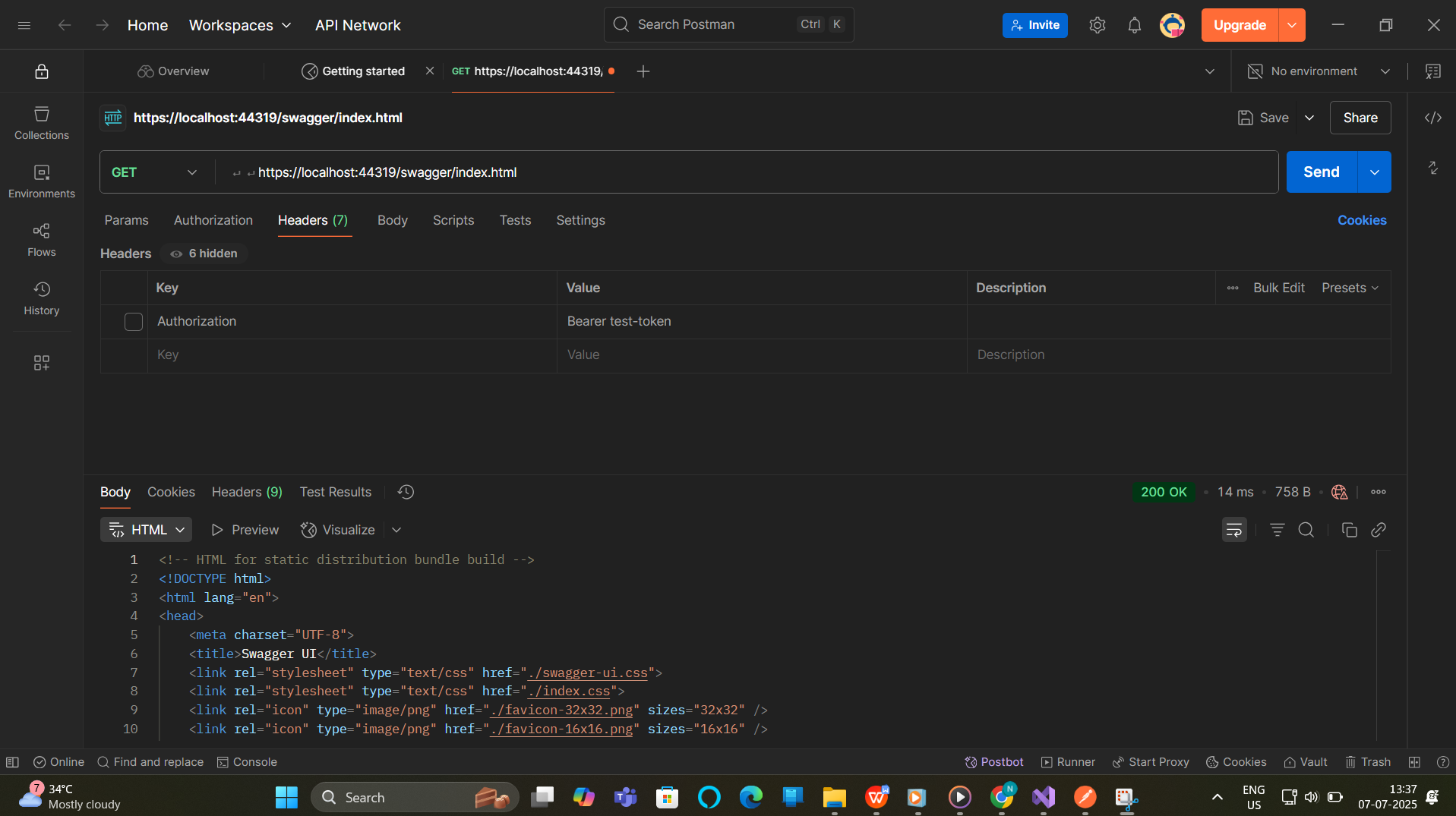
public int Id { get; set; }

public string Name { get; set; }

}

}

**OUTPUT:**



2)

EmployeeController.cs

using Microsoft.AspNetCore.Mvc;

using SwaggerDemoApi.Models;

namespace SwaggerDemoApi.Controllers

{

[ApiController]

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

public class EmployeeController : ControllerBase

{

private static List<Employee> employees = new List<Employee>();

// Constructor to initialize sample data

static EmployeeController()

{

employees = GetStandardEmployeeList();

}

// 🔹 GET: api/employee

[HttpGet]

[ProducesResponseType(typeof(List<Employee>), 200)] // ✅ Swagger annotation

public ActionResult<List<Employee>> Get()

{

return Ok(employees);

}

// 🔹 GET: api/employee/standard

[HttpGet("standard")]

public ActionResult<Employee> GetStandard()

{

return Ok(GetStandardEmployeeList().FirstOrDefault());

}

// 🔹 POST: api/employee

[HttpPost]

public ActionResult<Employee> Post([FromBody] Employee newEmp)

{

employees.Add(newEmp);

return CreatedAtAction(nameof(Get), new { id = newEmp.Id }, newEmp);

}

// 🔹 PUT: api/employee

[HttpPut]

public IActionResult Put([FromBody] Employee updatedEmp)

{

var existing = employees.FirstOrDefault(e => e.Id == updatedEmp.Id);

if (existing == null) return NotFound();

existing.Name = updatedEmp.Name;

existing.Salary = updatedEmp.Salary;

existing.Permanent = updatedEmp.Permanent;

existing.Department = updatedEmp.Department;

existing.Skills = updatedEmp.Skills;

existing.DateOfBirth = updatedEmp.DateOfBirth;

return NoContent();

}

// 🔐 Private method to generate default data

private static List<Employee> GetStandardEmployeeList()

{

return new List<Employee>

{

new Employee

{

Id = 1,

Name = "John Doe",

Salary = 50000,

Permanent = true,

Department = new Department { Id = 1, Name = "IT" },

Skills = new List<Skill>

{

new Skill { Id = 1, Name = "C#" },

new Skill { Id = 2, Name = "SQL" }

},

DateOfBirth = new DateTime(1990, 5, 23)

},

new Employee

{

Id = 2,

Name = "Jane Smith",

Salary = 60000,

Permanent = false,

Department = new Department { Id = 2, Name = "HR" },

Skills = new List<Skill>

{

new Skill { Id = 3, Name = "Recruitment" },

new Skill { Id = 4, Name = "Communication" }

},

DateOfBirth = new DateTime(1988, 3, 17)

}

};

}

}

}

