1. **Web Api using custom model class**

**Employee.cs**

**using System;**

**using System.Collections.Generic;**

**namespace MyFirstWebAPI.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; }**

**}**

**}**

**Department.cs**

**namespace MyFirstWebAPI.Models**

**{**

**public class Department**

**{**

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

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

**}**

**}**

**Skill.cs**

**namespace MyFirstWebAPI.Models**

**{**

**public class Skill**

**{**

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

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

**}**

**}**

**EmployeeController**

**using Microsoft.AspNetCore.Mvc;**

**using MyFirstWebAPI.Models;**

**using System.Collections.Generic;**

**using System;**

**namespace MyFirstWebAPI.Controllers**

**{**

**[ApiController]**

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

**public class EmployeeController : ControllerBase**

**{**

**private readonly List<Employee> \_employees;**

**public EmployeeController()**

**{**

**\_employees = GetStandardEmployeeList();**

**}**

**private List<Employee> GetStandardEmployeeList()**

**{**

**return new List<Employee>**

**{**

**new Employee**

**{**

**Id = 1,**

**Name = "John",**

**Salary = 50000,**

**Permanent = true,**

**DateOfBirth = new DateTime(1990, 1, 1),**

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

**Skills = new List<Skill> {**

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

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

**}**

**}**

**};**

**}**

**[HttpGet]**

**[ProducesResponseType(typeof(List<Employee>), 200)]**

**[ProducesResponseType(500)]**

**public ActionResult<List<Employee>> Get()**

**{**

**return \_employees;**

**}**

**[HttpPost]**

**public ActionResult<Employee> Post([FromBody] Employee employee)**

**{**

**\_employees.Add(employee);**

**return Ok(employee);**

**}**

**[HttpPut("{id}")]**

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

**{**

**return Ok("Employee updated.");**

**}**

**[HttpGet("standard")]**

**public ActionResult<Employee> GetStandrad()**

**{**

**return \_employees[0];**

**}**

**}**

**}**

**CustomAuthFilter**

**using Microsoft.AspNetCore.Mvc;**

**using Microsoft.AspNetCore.Mvc.Filters;**

**using System.Linq;**

**namespace MyFirstWebAPI.Filters**

**{**

**public class CustomAuthFilter : ActionFilterAttribute**

**{**

**public override void OnActionExecuting(ActionExecutingContext context)**

**{**

**var hasHeader = context.HttpContext.Request.Headers.TryGetValue("Authorization", out var token);**

**if (!hasHeader)**

**{**

**context.Result = new BadRequestObjectResult("Invalid request - No Auth token");**

**return;**

**}**

**if (!token.ToString().Contains("Bearer"))**

**{**

**context.Result = new BadRequestObjectResult("Invalid request - Token present but Bearer unavailable");**

**}**

**}**

**}**

**}**

**CustomExceptionFilter**

**using Microsoft.AspNetCore.Mvc;**

**using Microsoft.AspNetCore.Mvc.Filters;**

**using System;**

**using System.IO;**

**namespace MyFirstWebAPI.Filters**

**{**

**public class CustomExceptionFilter : IExceptionFilter**

**{**

**public void OnException(ExceptionContext context)**

**{**

**string logPath = Path.Combine(Directory.GetCurrentDirectory(), "Logs");**

**Directory.CreateDirectory(logPath);**

**File.AppendAllText(Path.Combine(logPath, "errorlog.txt"),**

**$"[{DateTime.Now}] Exception: {context.Exception.Message}{Environment.NewLine}");**

**context.Result = new ObjectResult("An unexpected error occurred")**

**{**

**StatusCode = 500**

**};**

**}**

**}**

**}**