**Boppudi Vivek(916243)**

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

Create a Custom class ‘Employee’ of the below defined structure

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; }

}

Create a new controller - EmployeeController with Read Write actions

Constructor: Create few records, HTTPGet, HTTPPost/HTTPPut

Create a Private method GetStandardEmployeeList that returns a List of Employee class. Invoke this method in the Get action method of the EmployeeController that was created in the previous step.

Public ActionResult<Employee> GetStandrad()

Modify the return type of the Get action method(without parameter) to return List of Employee class object

Add ProducesResponseType to the GET action method for Status code 200

Check the Swagger description for the GET method for success status code

**Code=>**

**Employee.cs:**

**using System;**

**using System.Collections.Generic;**

**using System.Linq;**

**using System.Threading.Tasks;**

**namespace WebApiDayHandson2.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 string Department { get; set; }**

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

**public DateTime DateOfBirth { 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 WebApiDayHandson2.Models;**

**namespace WebApiDayHandson2.Controllers**

**{**

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

**[ApiController]**

**public class EmployeesController : ControllerBase**

**{**

**private Employee[] emp = new Employee[]**

**{**

**new Employee { Id=1 , Name="vivek" , Salary=20000 , Permanent=true, Department="CSE",Skills="HTML" ,DateOfBirth=new DateTime(1998,08,07) },**

**new Employee { Id=2 , Name="pavan" , Salary=15000 , Permanent=true, Department="ECE",Skills="CSS" ,DateOfBirth=new DateTime(1999,11,09) } ,**

**new Employee { Id=3 , Name="hari" , Salary=10000 , Permanent=false, Department="EEE",Skills="JAVA" ,DateOfBirth=new DateTime(1997,02,12) } ,**

**};**

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

**{**

**return emp;**

**}**

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

**[CustomAuthFilter]**

**[HttpGet]**

**public IEnumerable<Employee> Get()**

**{**

**return GetStandardEmployeeList();**

**}**

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

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

**[ProducesResponseType(StatusCodes.Status200OK)]**

**[ProducesResponseType(StatusCodes.Status404NotFound)]**

**public IActionResult GetById(int id)**

**{**

**var prod = emp.FirstOrDefault((p) => p.Id == id);**

**if (prod == null)**

**{**

**return NotFound();**

**}**

**return Ok(prod);**

**}**

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

**[HttpPost]**

**public void Post([FromBody] string value)**

**{**

**}**

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

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

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

**{**

**}**

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

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

**public void Delete(int id)**

**{**

**}**

**}**

**}**

**CustomAuthFilter.cs**

**using Microsoft.AspNetCore.Mvc;**

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

**using Microsoft.Extensions.Primitives;**

**using System;**

**using System.Collections.Generic;**

**using System.Diagnostics;**

**using System.Linq;**

**using System.Threading.Tasks;**

**namespace WebApiDayHandson2.Models**

**{**

**public class CustomAuthFilter : ActionFilterAttribute**

**{**

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

**{**

**if (context.HttpContext.Request.Query.ContainsKey("Authorization") && context.HttpContext.Request.Query["Authorization"] == "true")**

**{**

**context.Result = new UnauthorizedResult();**

**}**

**else**

**{**

**base.OnActionExecuting(context);**

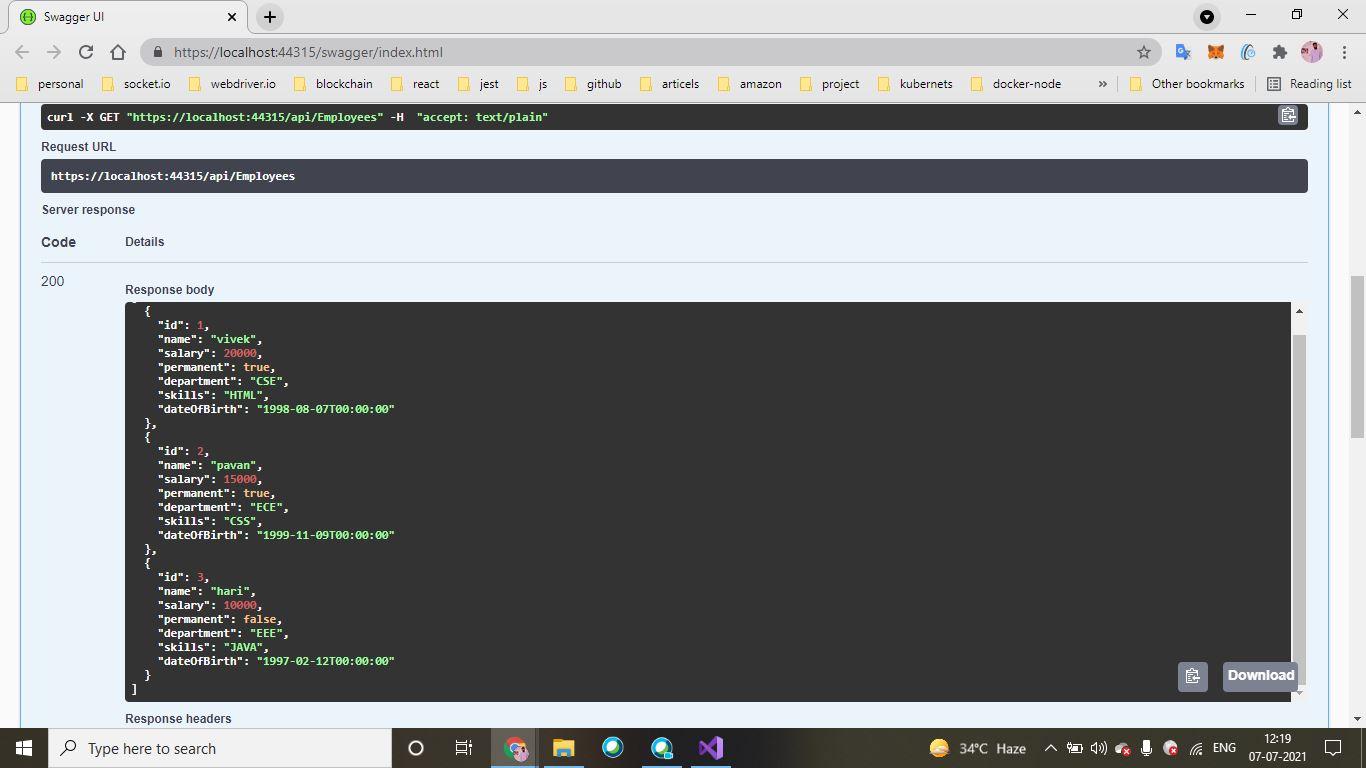
**}**

**}**

**}**

**}**

**Output:**

****

**2)**

**CustomExceptionFilter.cs**

**using System;**

**using System.Collections.Generic;**

**using System.IO;**

**using System.Linq;**

**using System.Threading.Tasks;**

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

**namespace WebApiDayHandson2.Models**

**{**

**public class CustomExceptionFilter : ExceptionFilterAttribute, IExceptionFilter**

**{**

**public override void OnException(ExceptionContext context)**

**{**

**string fileName = @"C:\Temp\customexception.txt";**

**using (StreamWriter writer = new StreamWriter(fileName))**

**{**

**writer.WriteLine(context.Exception.Message);**

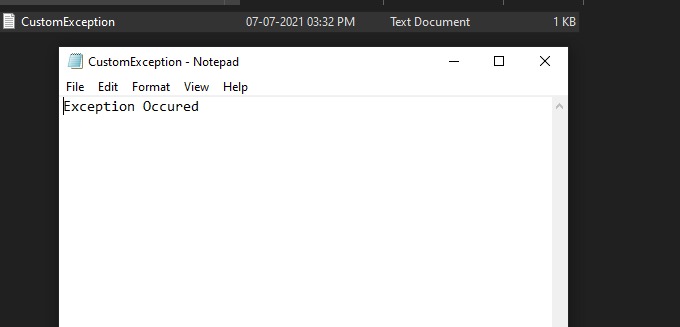
**}**

**}**

**}**

**}**

**Output:**

****