# Part 3: Custom Model & Filters - Hands-on Solution

WEEK4\_EXERCISE 3\_CUSTOM MODEL & FILTERS

## Objectives

* • Return list of custom class entities via Action methods.
* • Model class creation: Employee, Department, Skill.
* • Use AllowAnonymous and HttpGet action methods.
* • Explain usage of [FromBody] in POST.
* • Implement CustomAuthFilter (ActionFilterAttribute).
* • Implement CustomExceptionFilter (IExceptionFilter).

## 1. Model Classes

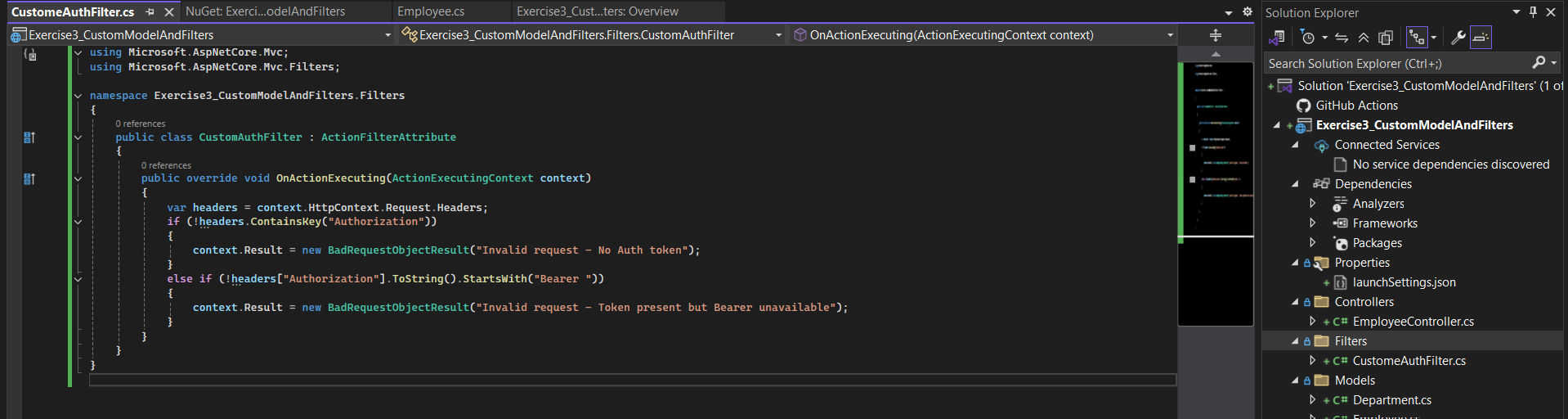
Under Models/ folder, create:

1. Department.cs
2. Skill.cs
3. Employee.cs

## 3 folders Created under Models

## 2. CustomAuthFilter

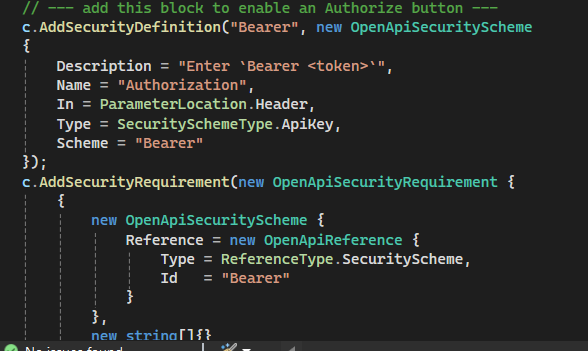
Under Filters/, create CustomAuthFilter inheriting ActionFilterAttribute.



## 3. Register Filter & Swagger Security in Program.cs

In Program.cs:

* • builder.Services.AddScoped<CustomAuthFilter>();
* • options.Filters.Add<CustomAuthFilter>(); in AddControllers
* • Add Swagger security definition for Bearer token



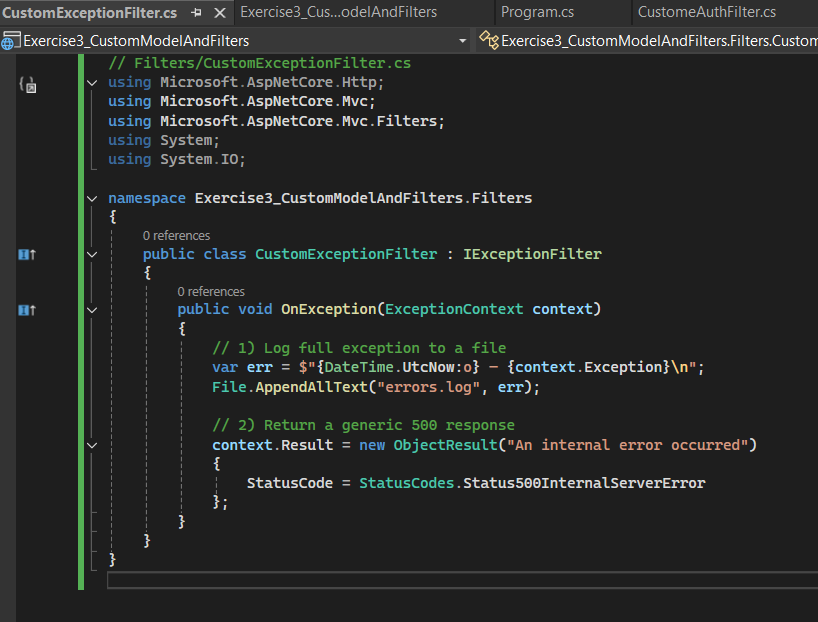
## 4. EmployeeController

Controller demonstrates GET, POST, PUT, DELETE with filters.



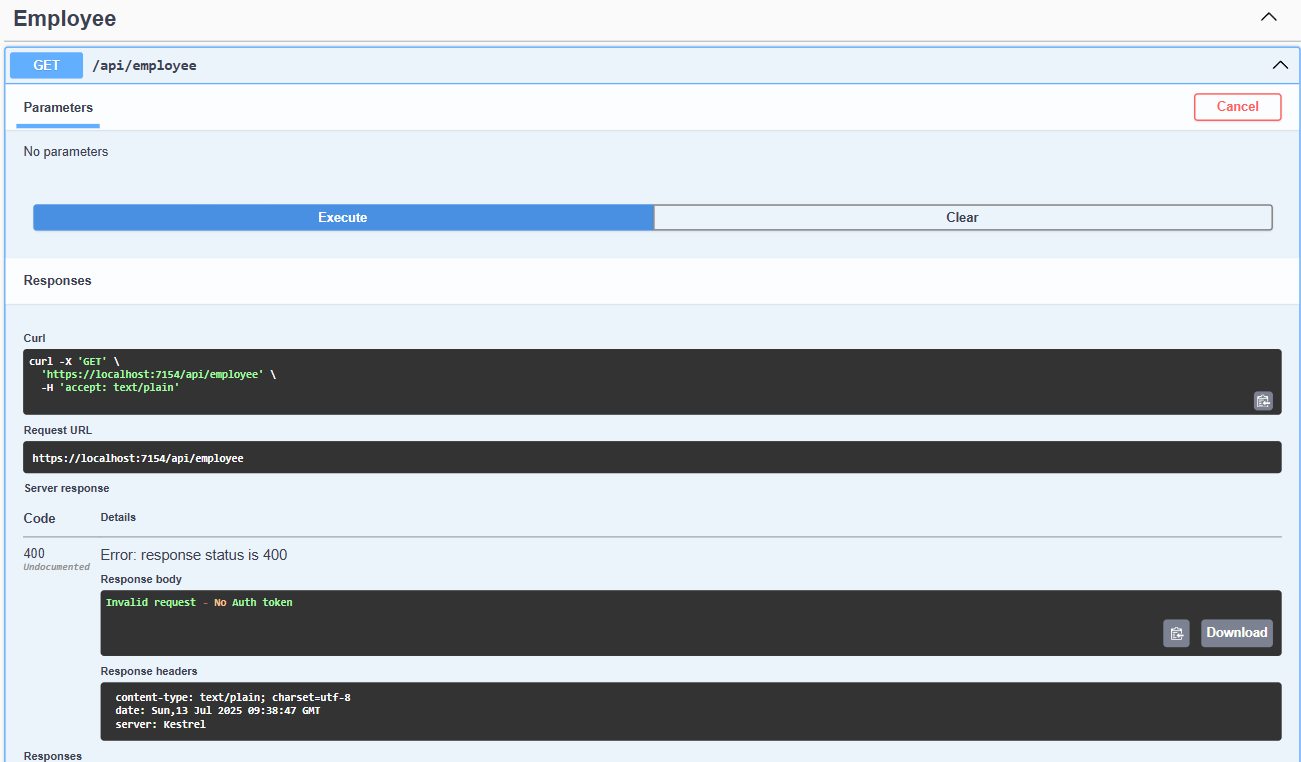
## 5. CustomExceptionFilter

Under Filters/, create CustomExceptionFilter implementing IExceptionFilter.

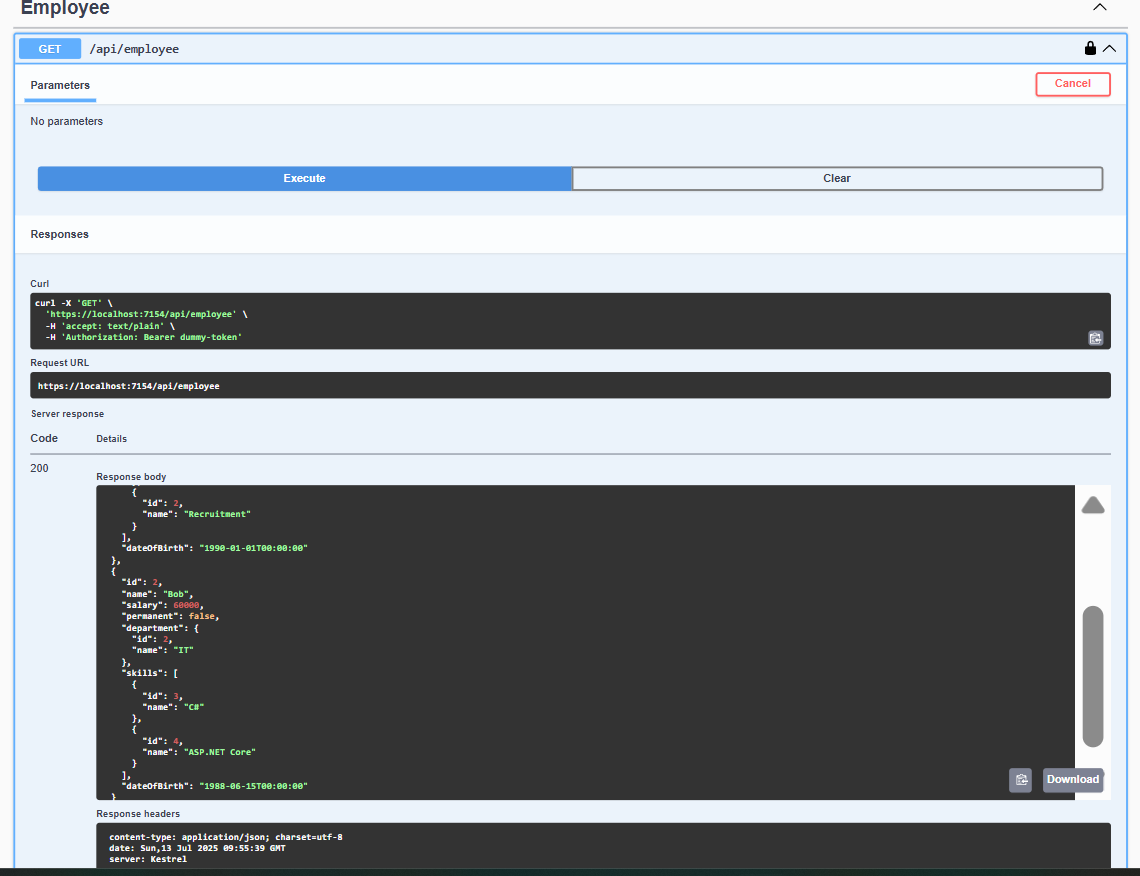


## 6. Testing Filters in Swagger UI

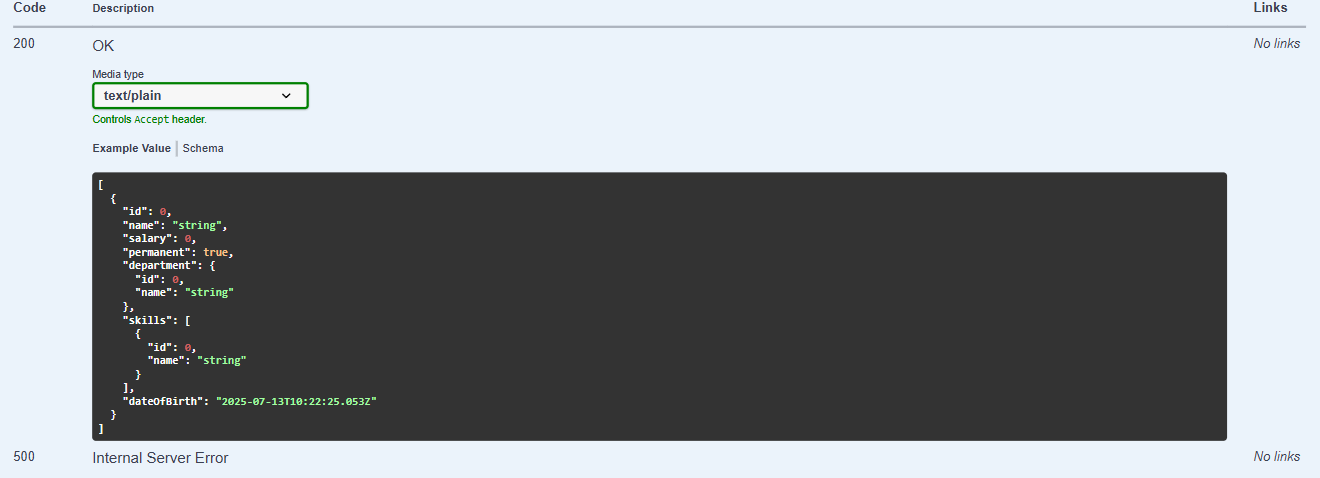
* - 400 Bad Request without Auth



* - 200 OK with Auth header



* - 500 Internal Error on exception



* - Logged exception in errors.log

