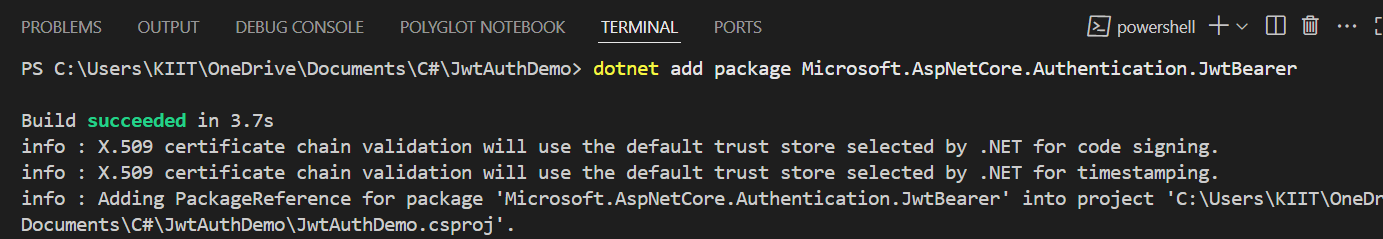
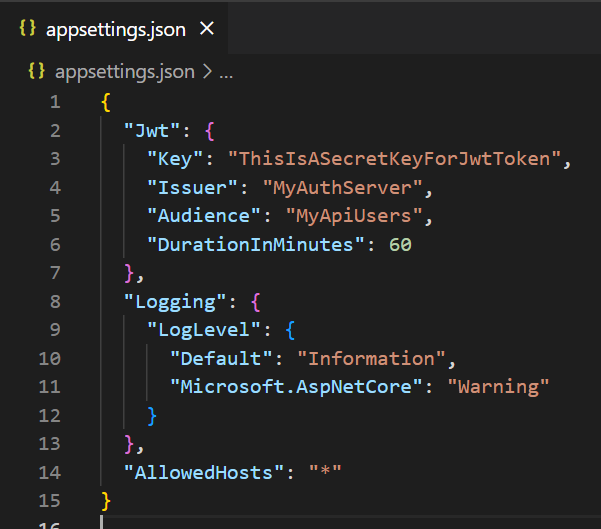
WEEK 5 MANDATORY HANDS-ON QUESTIONS

1. Implementing JWT authentication in ASP.NET Core Web Api.

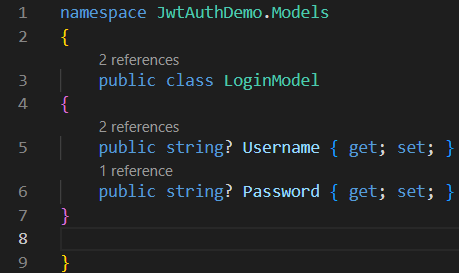
Step 1: Creating a new ASP.NET core web Api and adding the required packages.



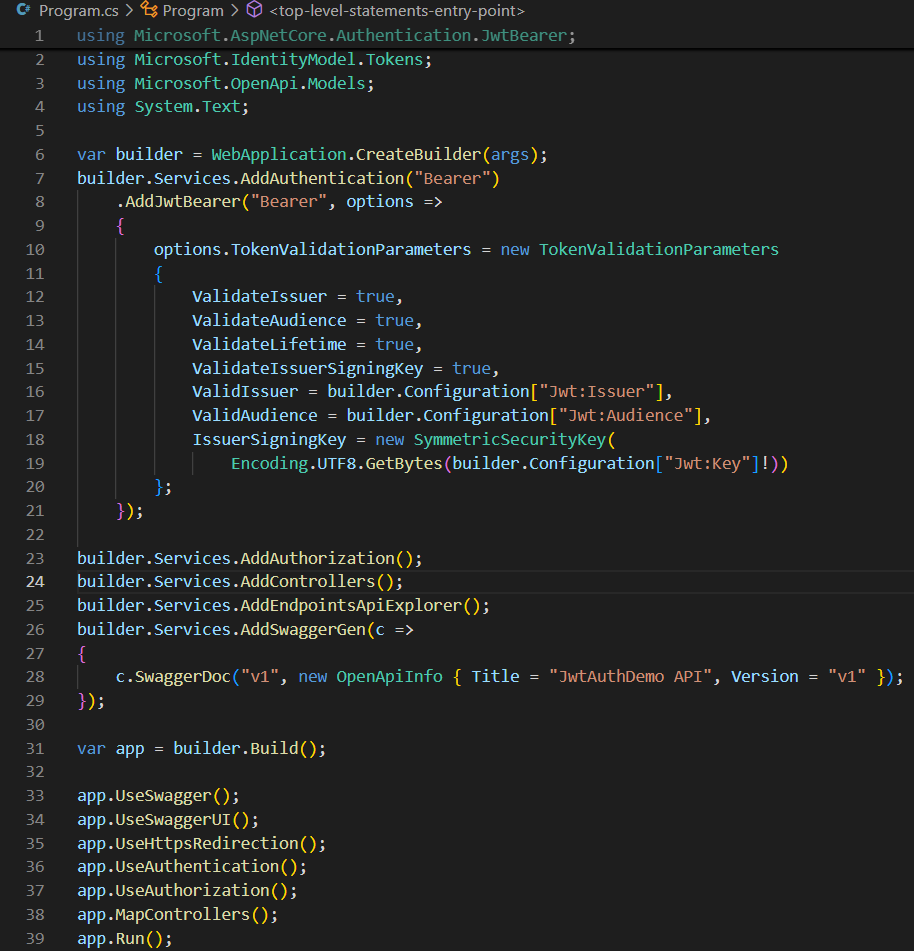
Step 2: Configuring and adding JWT configuration in appsettings.json



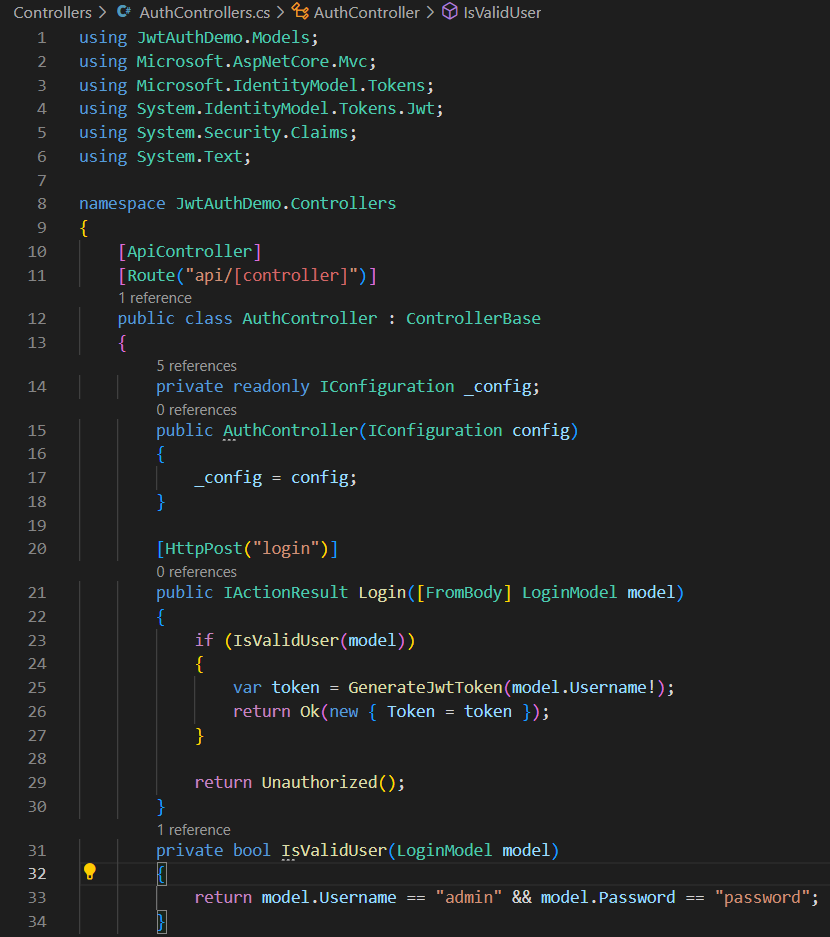
Step 3: Creating the Loginmodel and adding User model

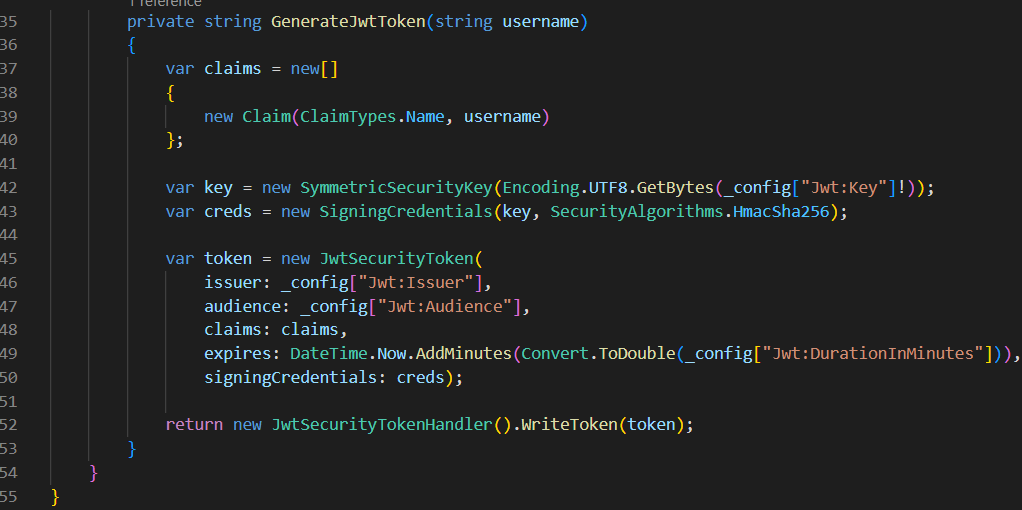


Step 4: Updating program.cs for JWT configuration

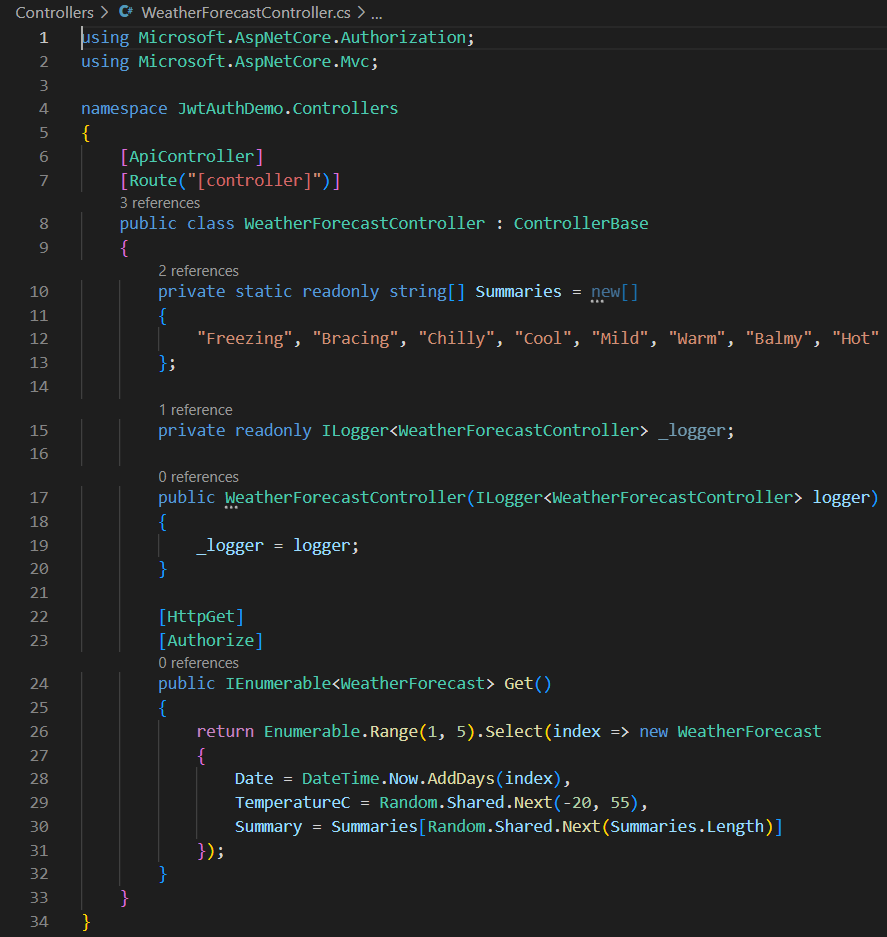


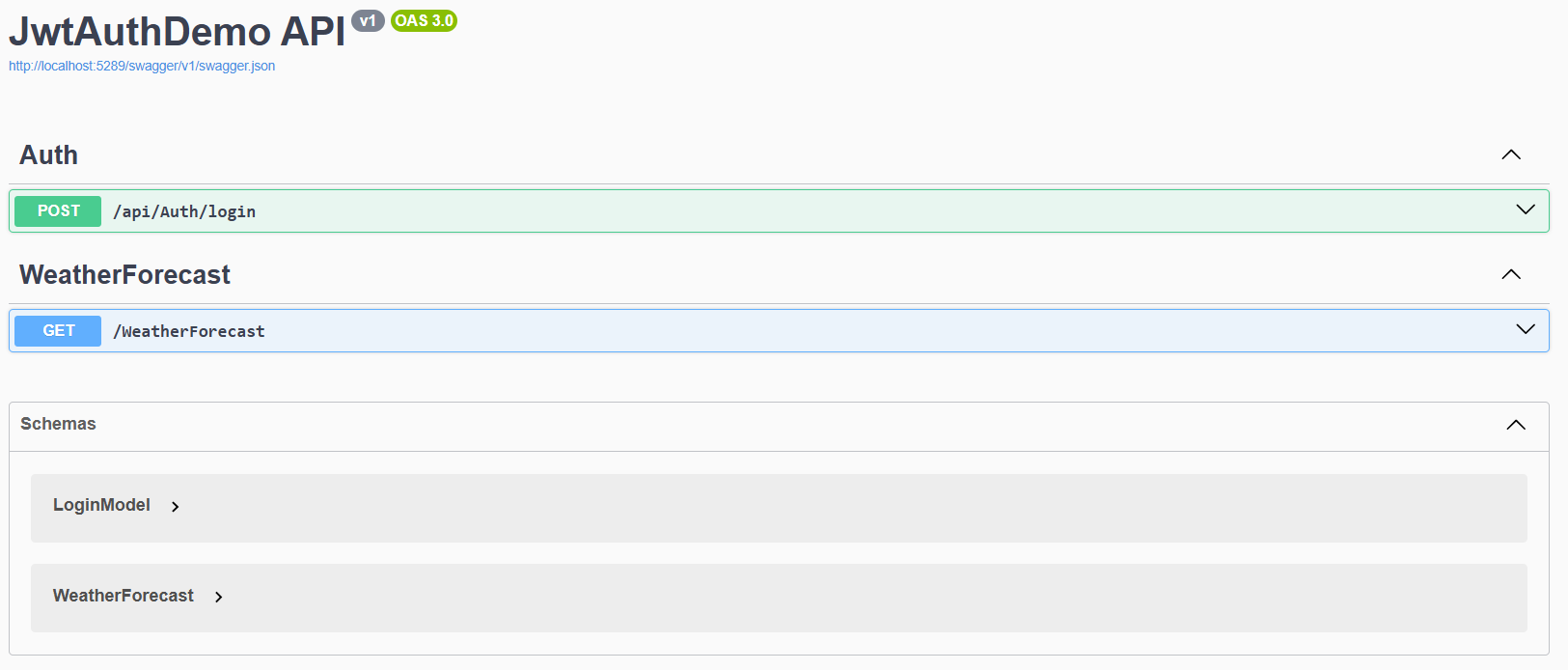
Step 5: Creating AuthControllers.cs

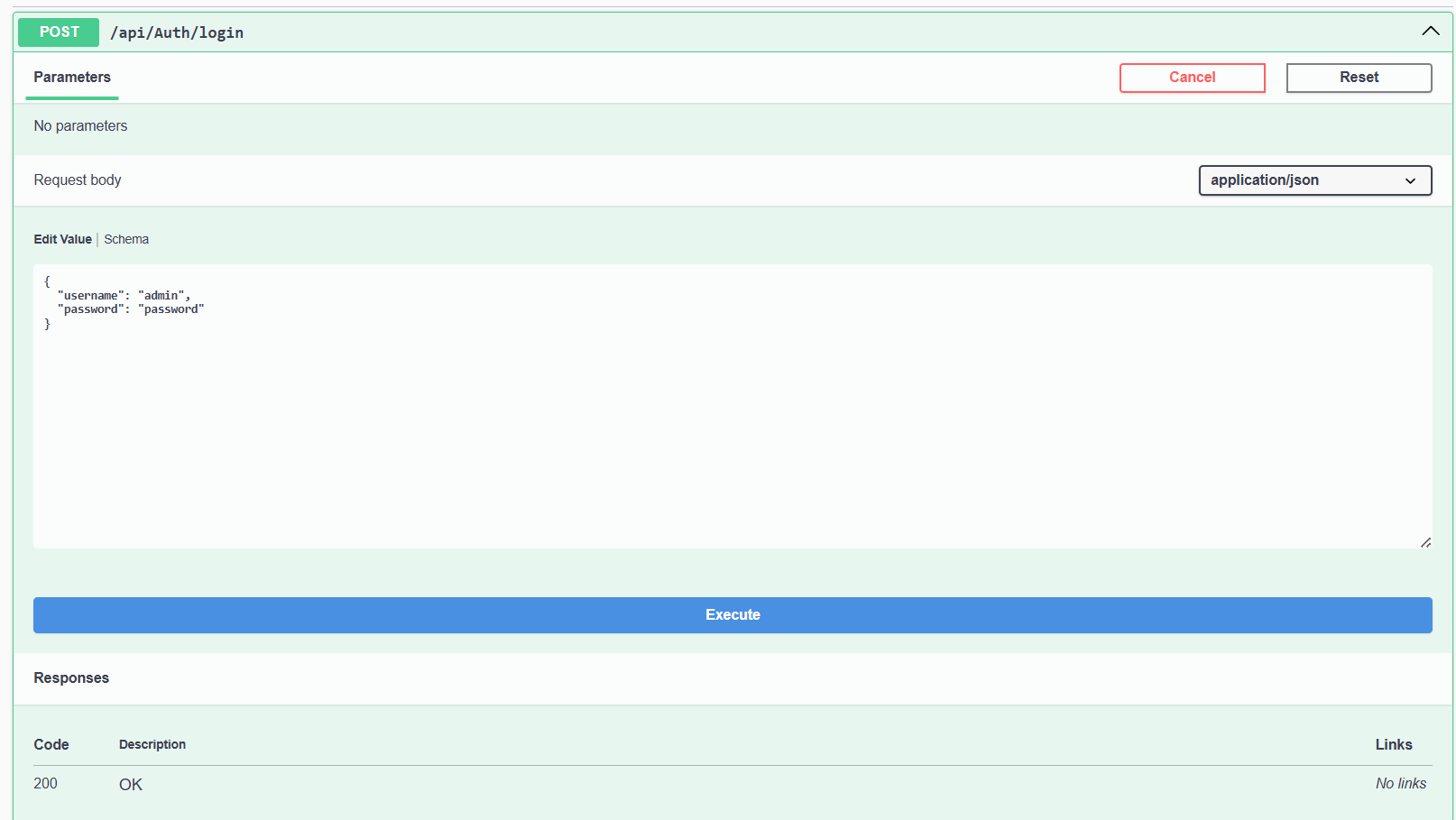




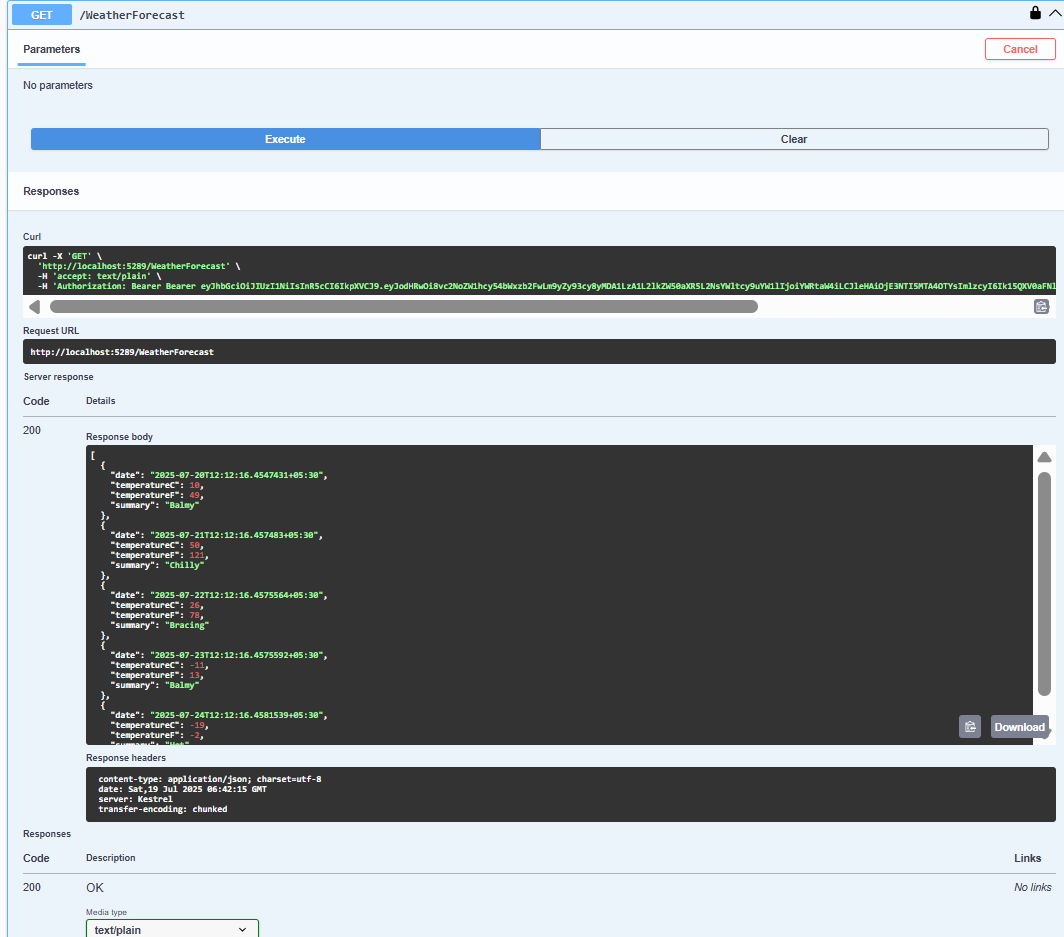
Step 6: Creating a protected controller



Now test the output on swagger

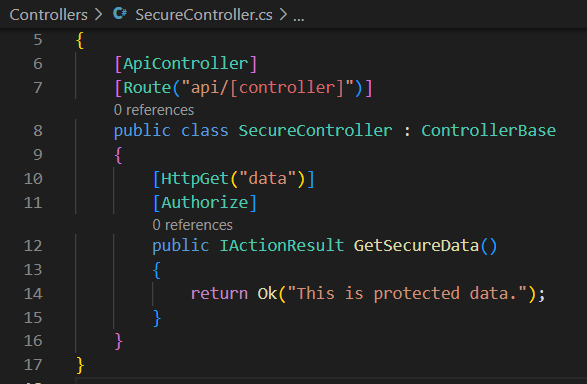


Testing /weatherforecast

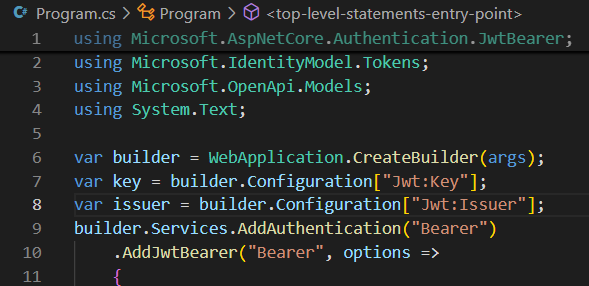


1. Securing the API endpoint using JWT.

Step 1: Creating a securecontroller.cs inside controller folder

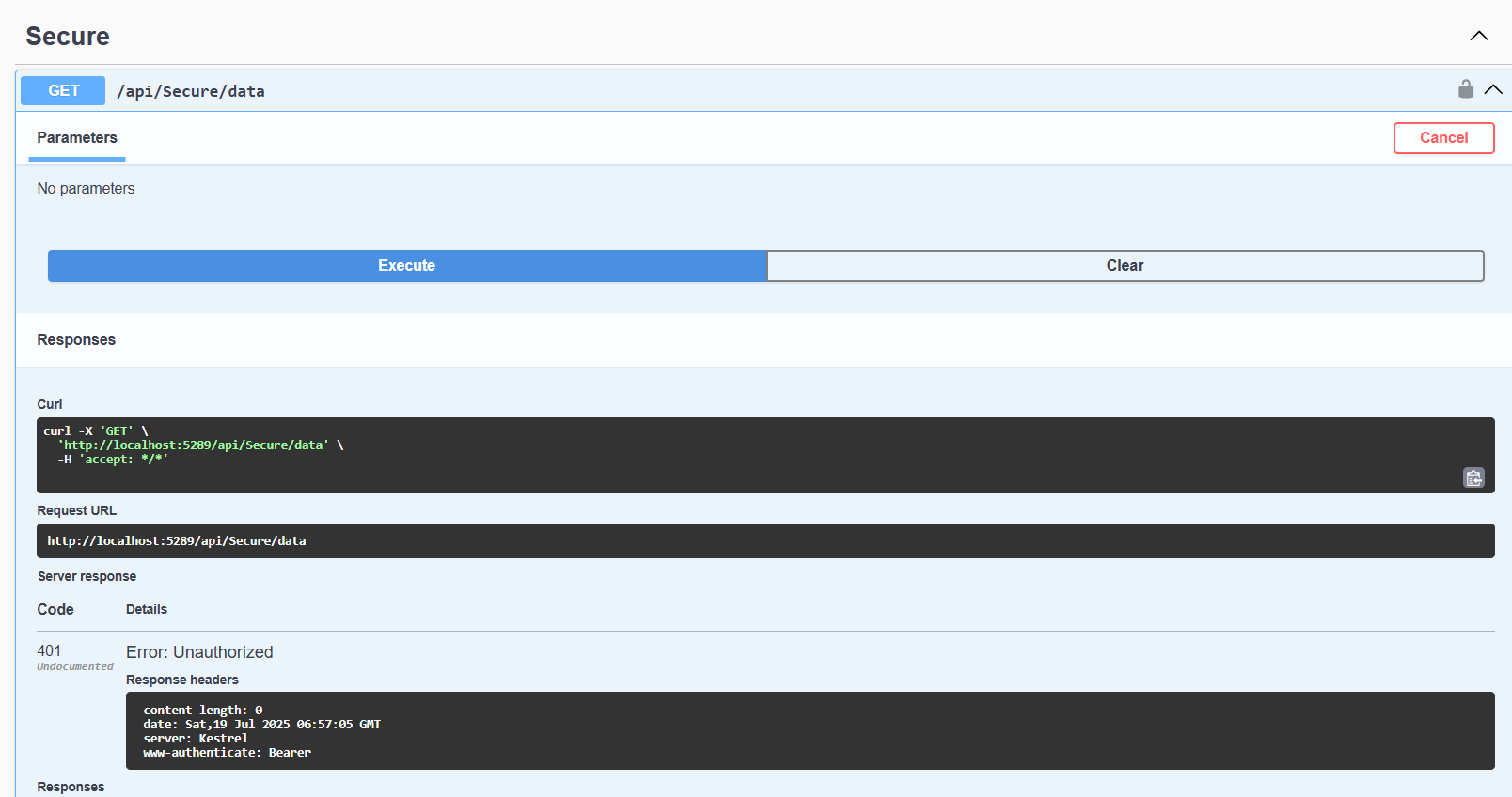


Step 2: Ensuring JWT authentication is registered in program.cs

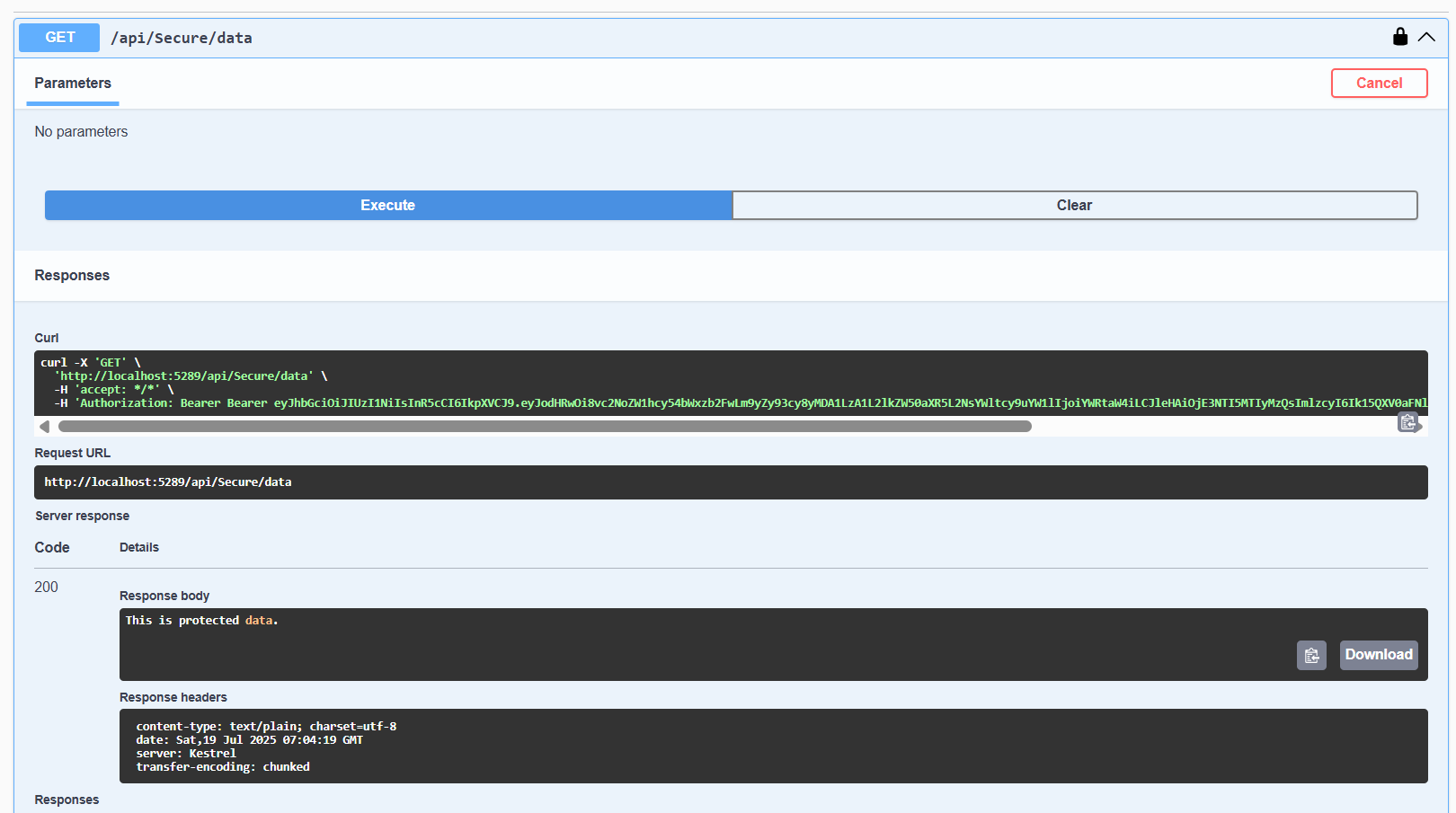


Now testing the access with or without the token in swagger

1. Without token

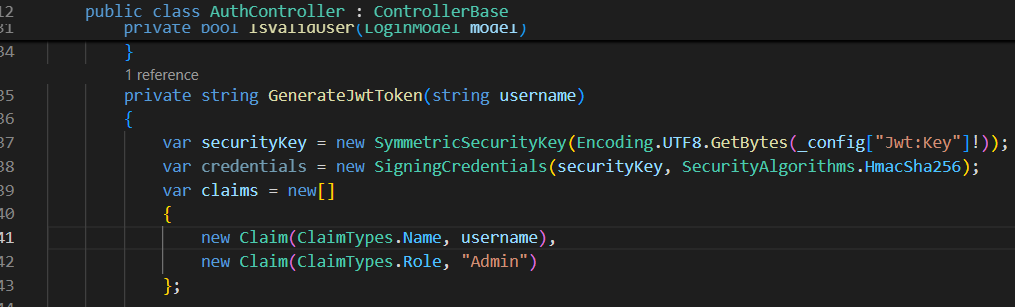


1. With token

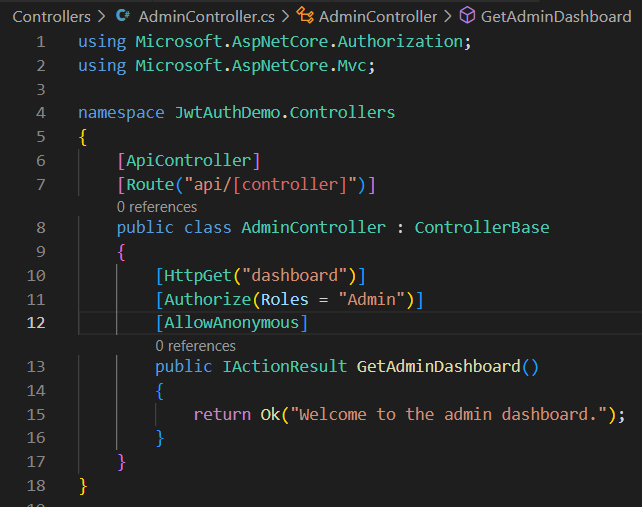


1. Adding a role-based authorization which will only allow users with “Admin” role to access the endpoints.

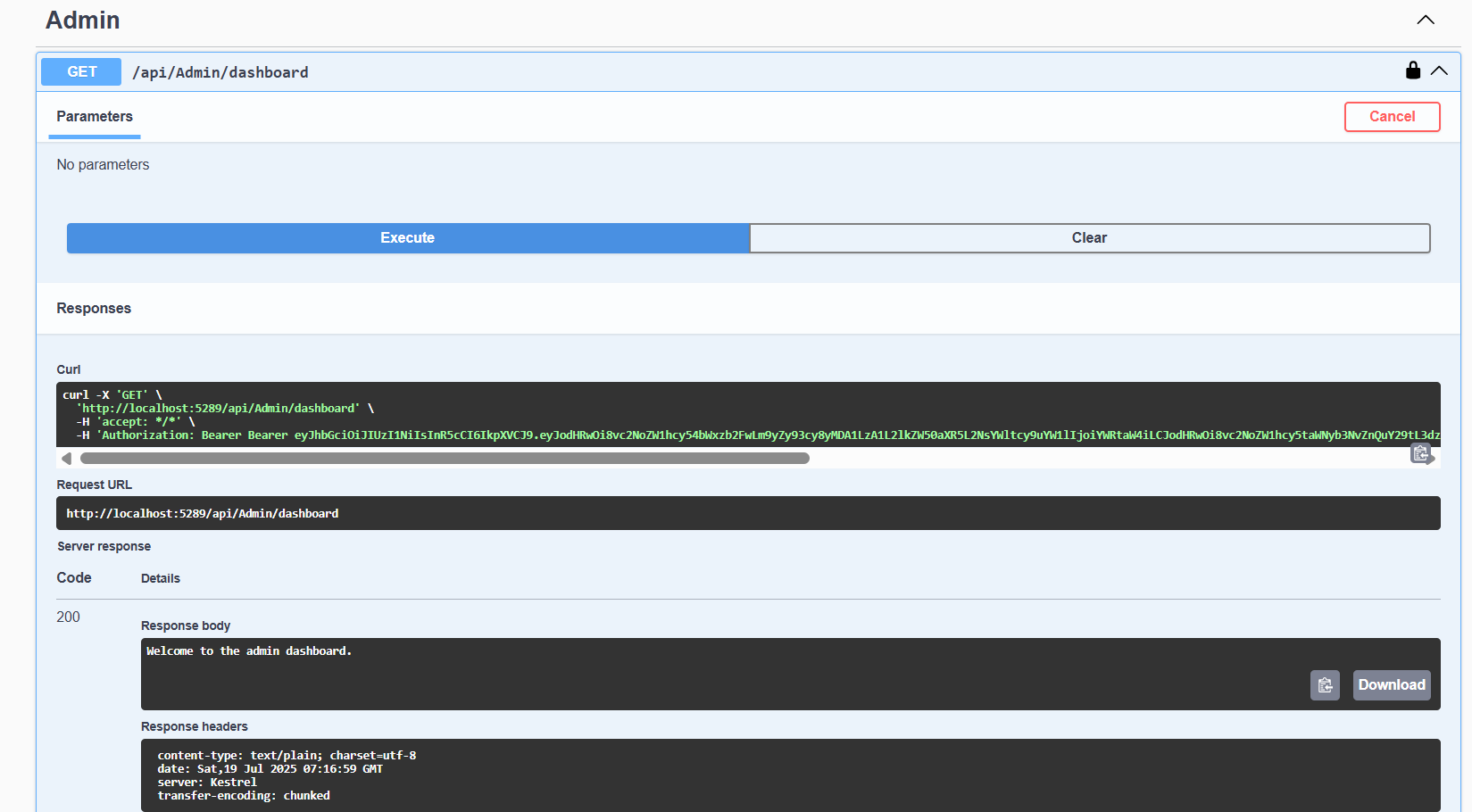
Step 1: Updating the GenerateJwtToken() to include the role claim



Step 2: Creating admincontroller.cs with role authorization



Now testing on swagger



1. Validating JWT token expiry and handle unauthorized access gracefully.

