* **Explain CORS enablement for Web API access for local application**
  + **What is CORS?, How to enable CORS thru Startup.cs, Install Cors nuget package to Web API application**

**Answer-->**

## What is CORS?

**CORS (Cross-Origin Resource Sharing)** is a browser security feature that restricts **web applications running at one origin (domain)** from accessing resources on a **different origin** (domain, port, or protocol).

For example, a React app at http://localhost:3000 trying to call a Web API at http://localhost:5000 would be blocked **unless CORS is enabled**.

## Why is CORS Important in Web API?

By default, browsers **block cross-origin calls** for security. Enabling CORS in Web API lets you allow specific applications (like local frontends) to access your backend API.

**How to Enable CORS in ASP.NET Core**

### 1. Install CORS NuGet Package (if needed)

For .NET 5+, **CORS is included by default**.  
But in earlier versions:

Install-Package Microsoft.AspNetCore.Cors

### 2. Modify Startup.cs to Enable CORS

#### a) Add the CORS service in ConfigureServices:

public void ConfigureServices(IServiceCollection services)

{

services.AddControllers();

// Add CORS

services.AddCors(options =>

{

options.AddPolicy("AllowLocalhost3000",

builder =>

{

builder.WithOrigins("http://localhost:3000") // React/Angular frontend

.AllowAnyHeader()

.AllowAnyMethod();

});

});

}

#### b) Enable the CORS middleware in Configure:

public void Configure(IApplicationBuilder app, IWebHostEnvironment env)

{

if (env.IsDevelopment())

{

app.UseDeveloperExceptionPage();

}

app.UseRouting();

// Enable CORS middleware

app.UseCors("AllowLocalhost3000");

app.UseAuthorization();

app.UseEndpoints(endpoints =>

{

endpoints.MapControllers();

});

}

### 3. Apply CORS Policy (Optional per-controller/per-method)

we can also apply it **per controller**:

[ApiController]

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

[EnableCors("AllowLocalhost3000")]public class StudentController : ControllerBase

{

// API actions

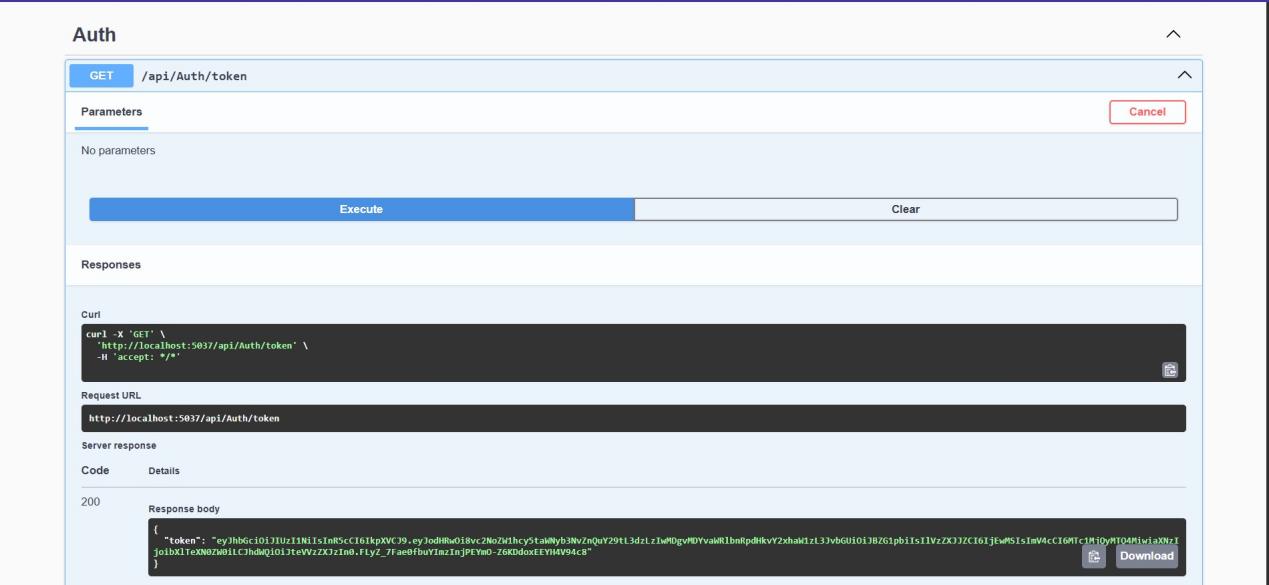
}

## How It Works

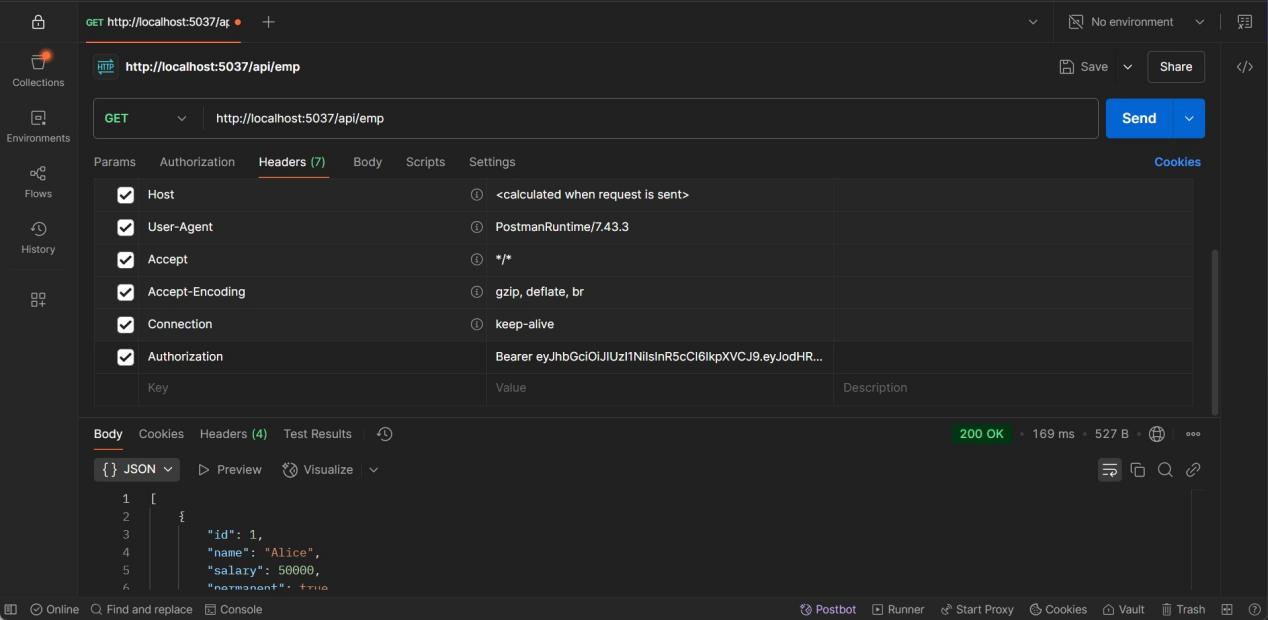
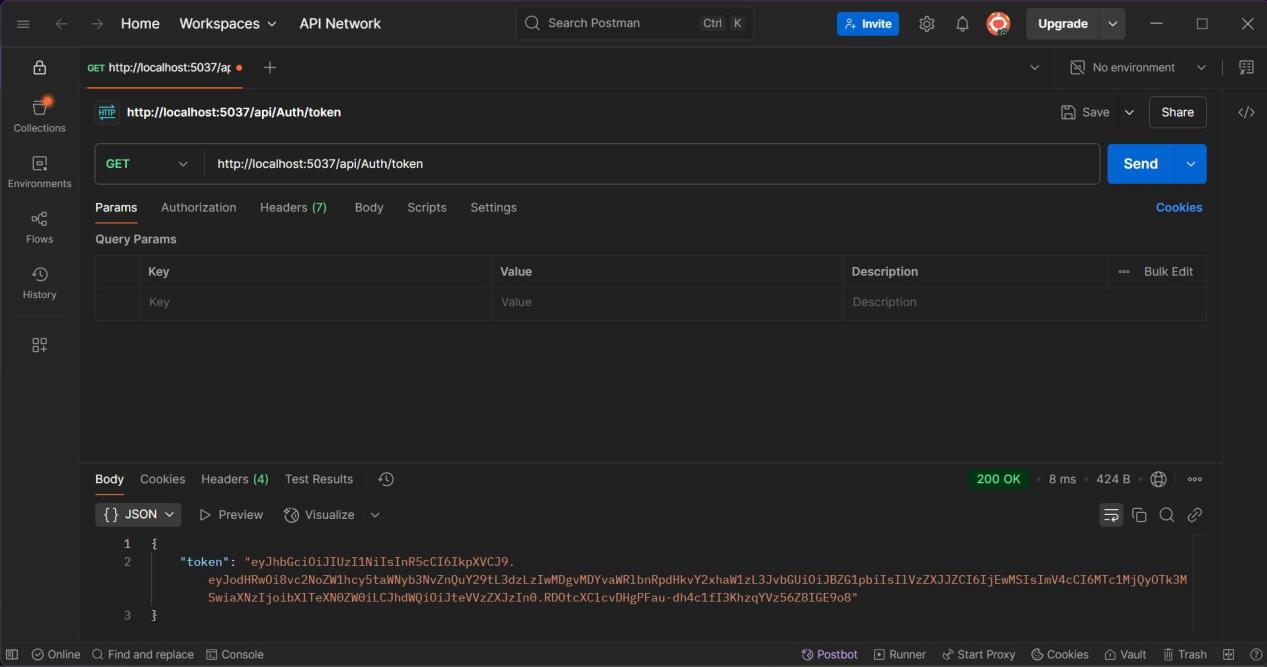
When a frontend app (e.g., React) calls your API with JavaScript, the browser sends a **preflight OPTIONS request**. If the response from your API doesn't have the proper Access-Control-Allow-Origin header, the request will be **blocked**.

**Output:**

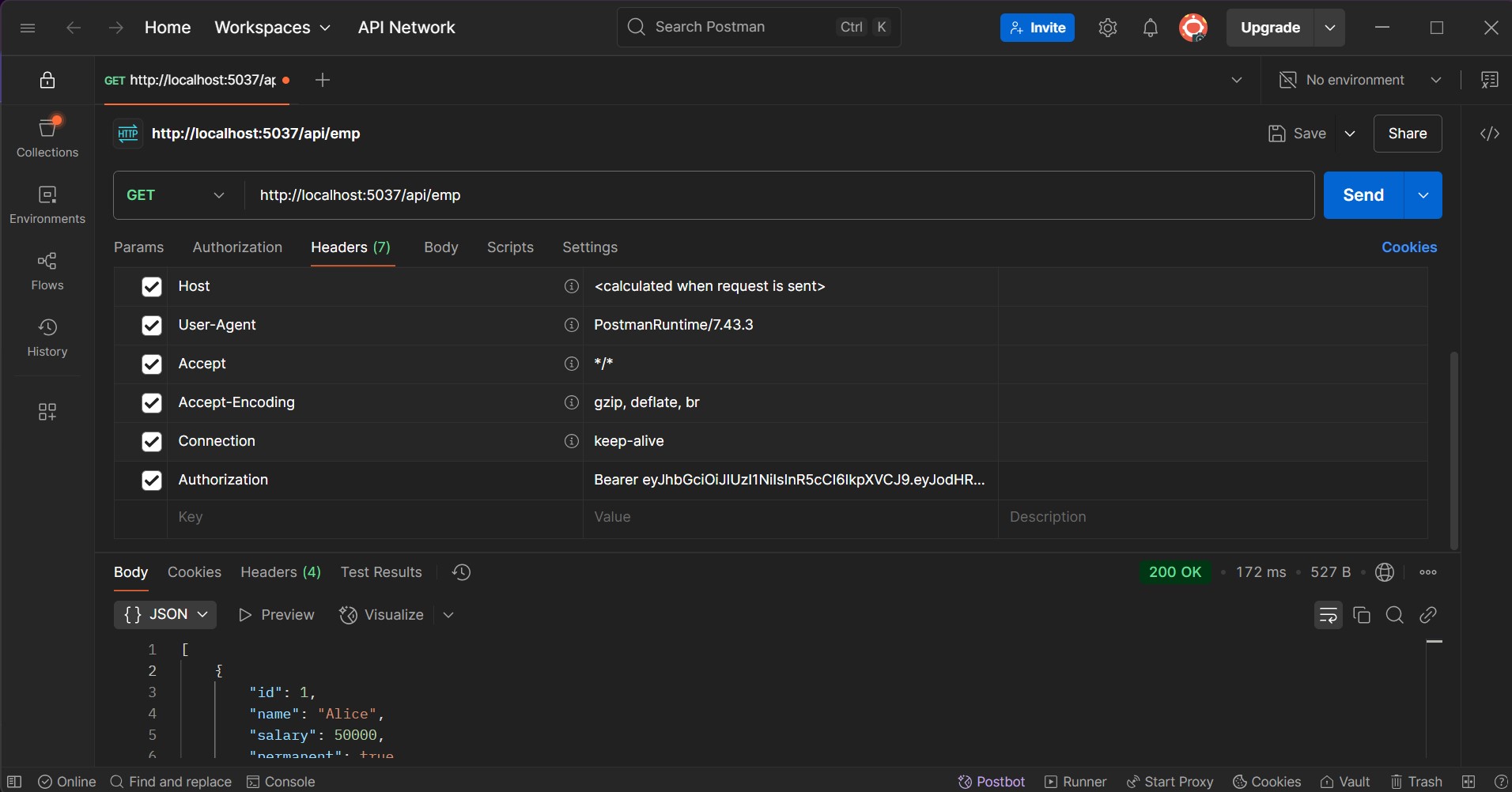
**1)**



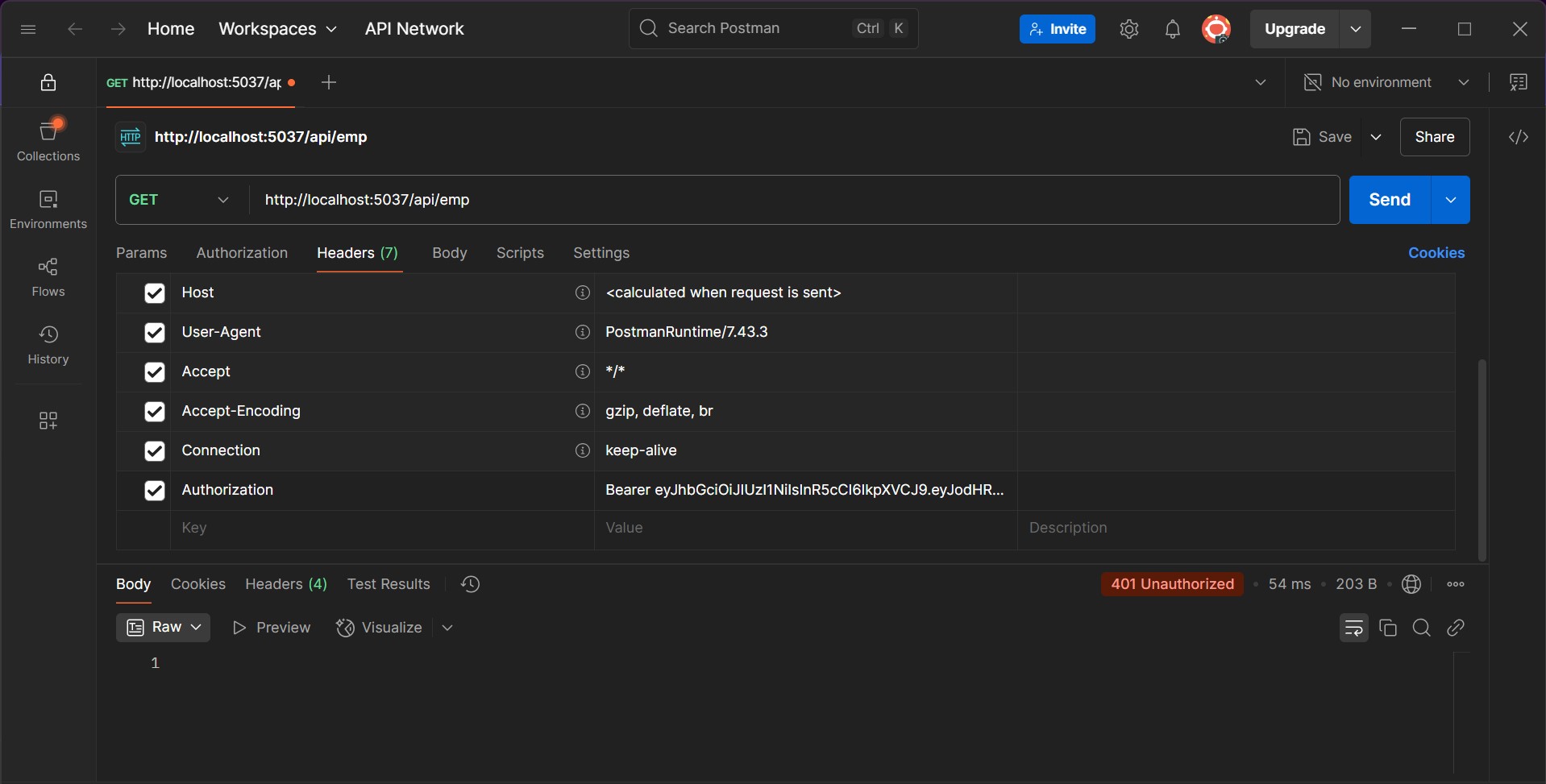
**2)**

****

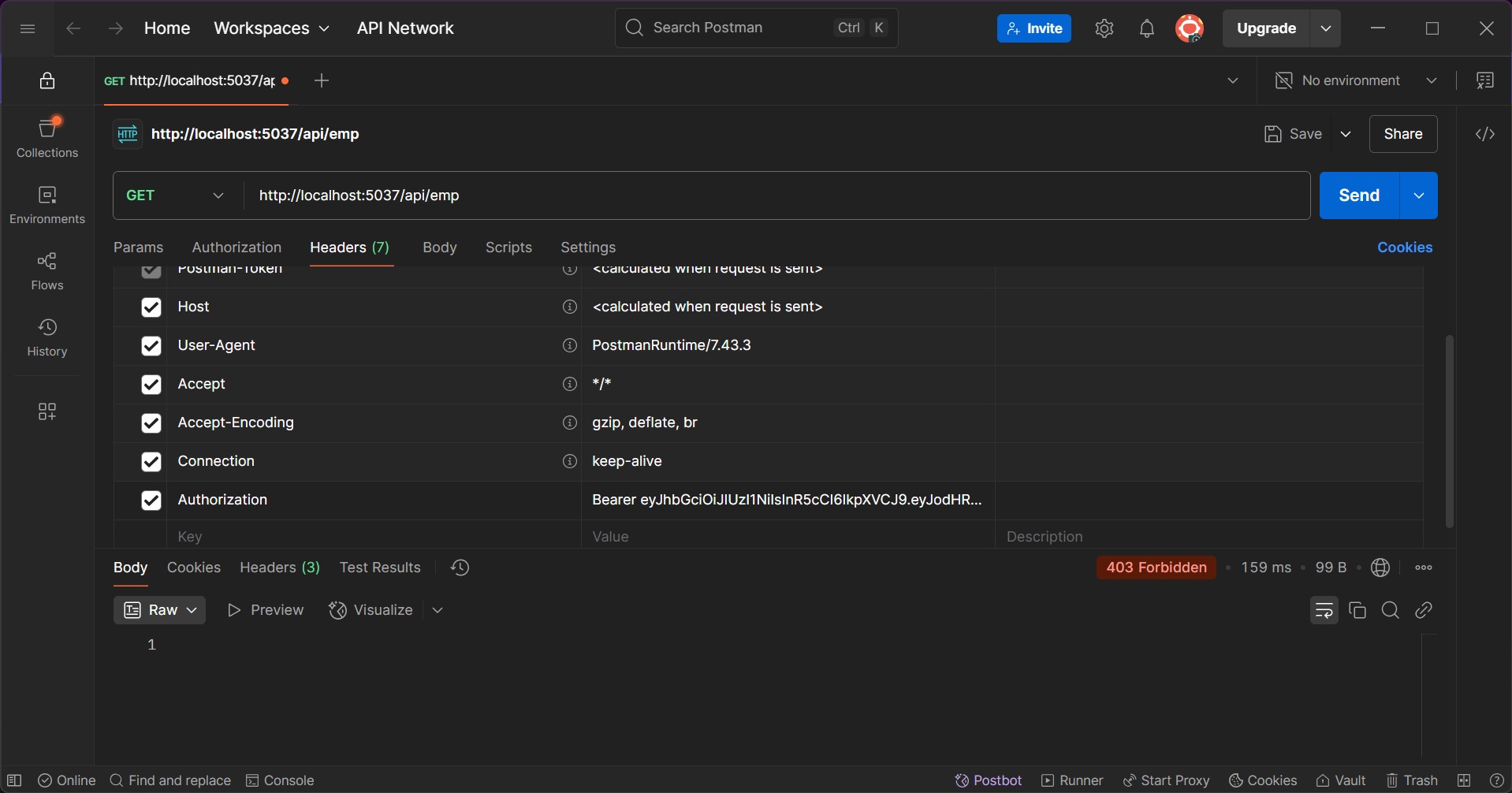
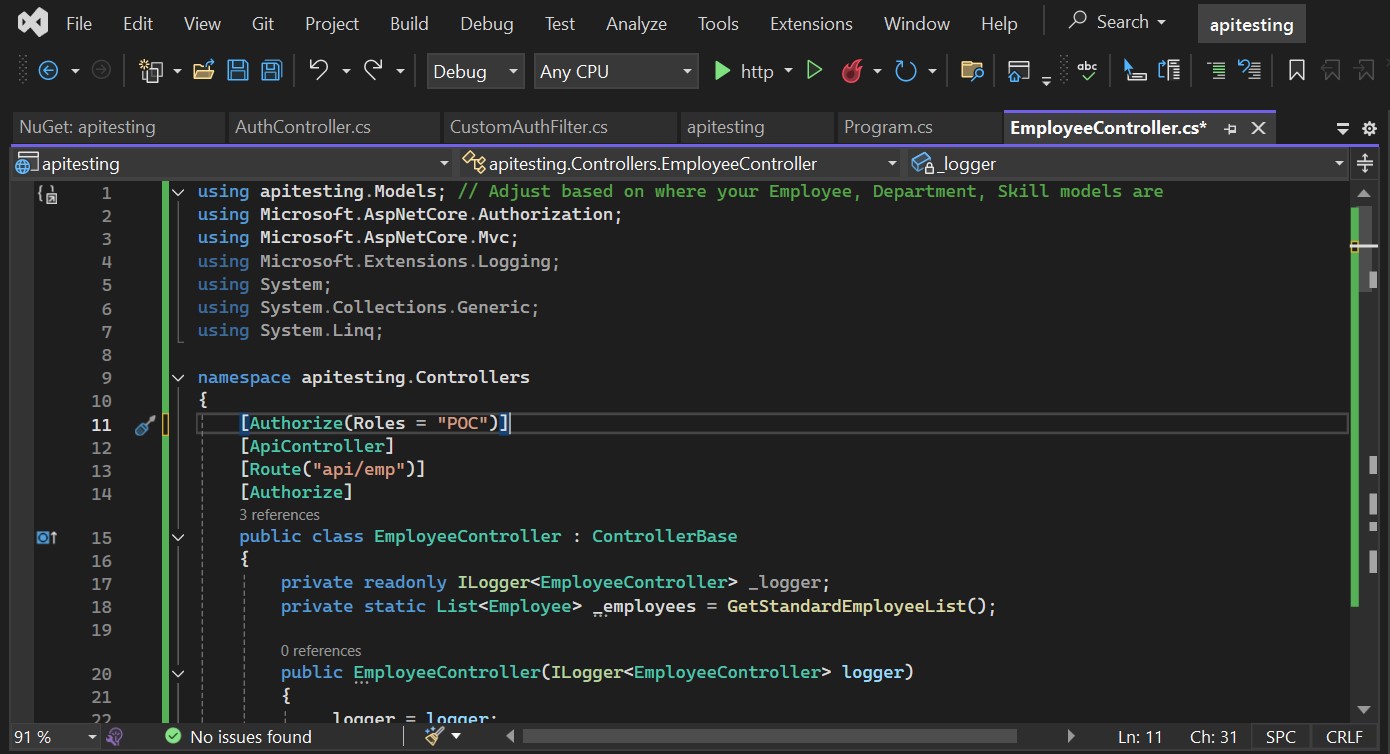
**3)**

****

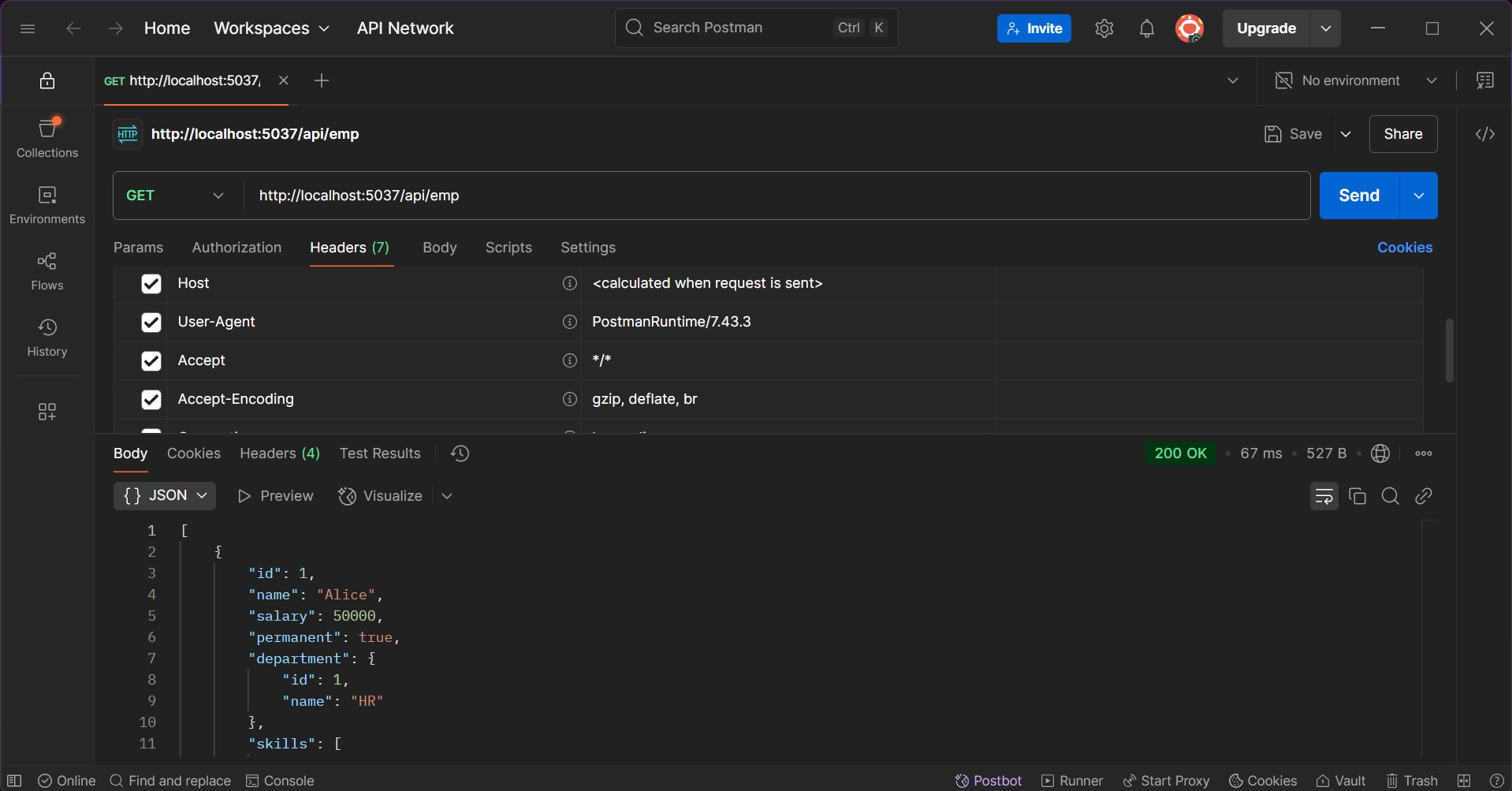
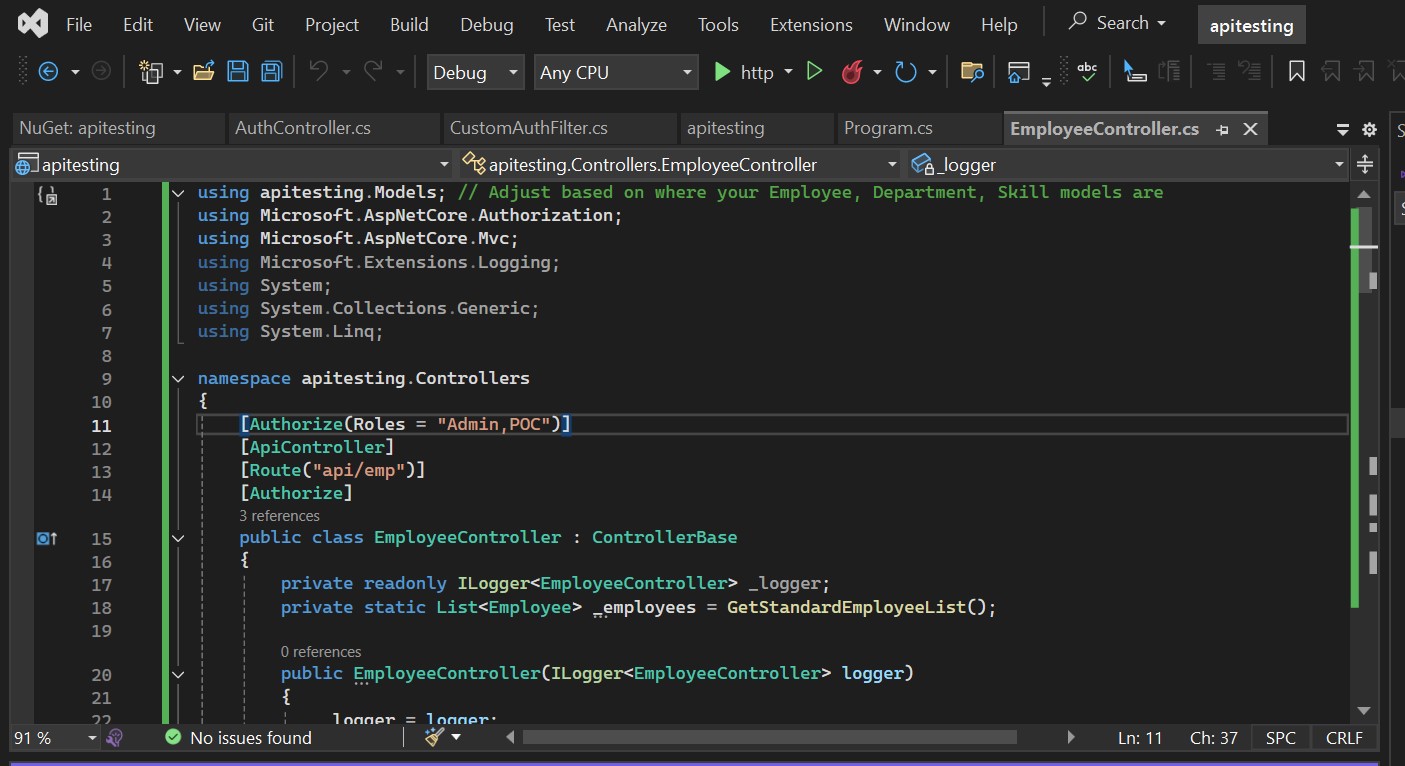
**After 2 mins:**

****

**4)Using POC**

****

**Using POC+ADMIN**

****