**WEEK 5**

1. **Microservices – JWT**

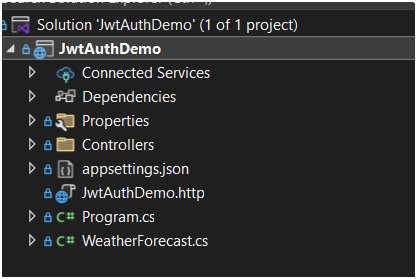
**Exercise1:Implement JWT Authentication in ASP.NET Core Web API**

**Scenario:**

**You are building a microservice that requires secure login. You need to implement JWT-based authentication**

**Step-by-Step in Visual Studio**

1. **Create a new ASP.NET Core Web API project**
   * **Open Visual Studio → Create a new project**
   * **Choose ASP.NET Core Web API → Click Next**
   * **Name: JwtAuthDemo → Click Create**
   * **Select .NET 6.0 or .NET 7.0, uncheck “Use controllers (Minimal API)” if checked → Click Create**

****

1. **Add the NuGet package**

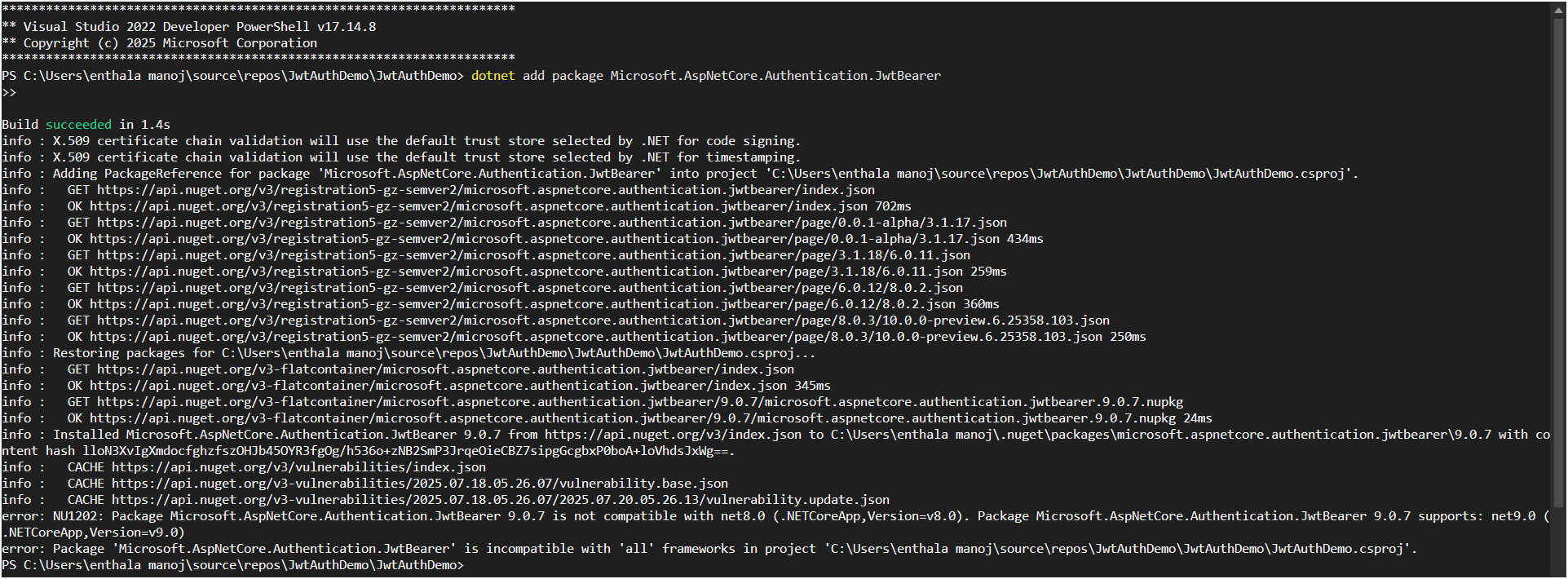
**Tools > NuGet Package Manager > Package Manager Console**

**Run:**

**bash**

**CopyEdit**

**dotnet add package Microsoft.AspNetCore.Authentication.JwtBearer**

**3.Add configuration in appsettings.json**

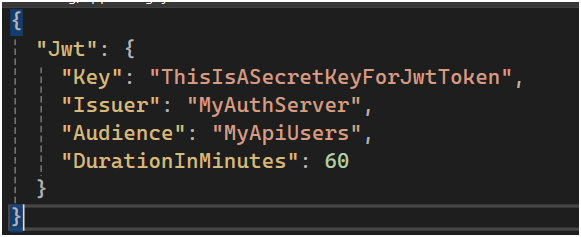
**{**

**"Jwt":**

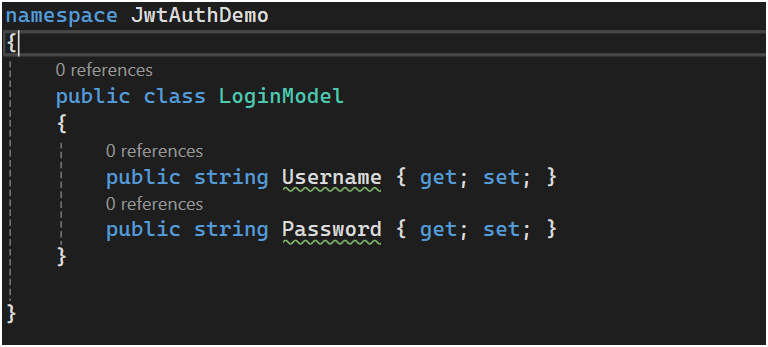
**{**

**"Key": "ThisIsASecretKeyForJwtToken", "Issuer": "MyAuthServer", "Audience": "MyApiUsers", "DurationInMinutes": 60**

**} }**

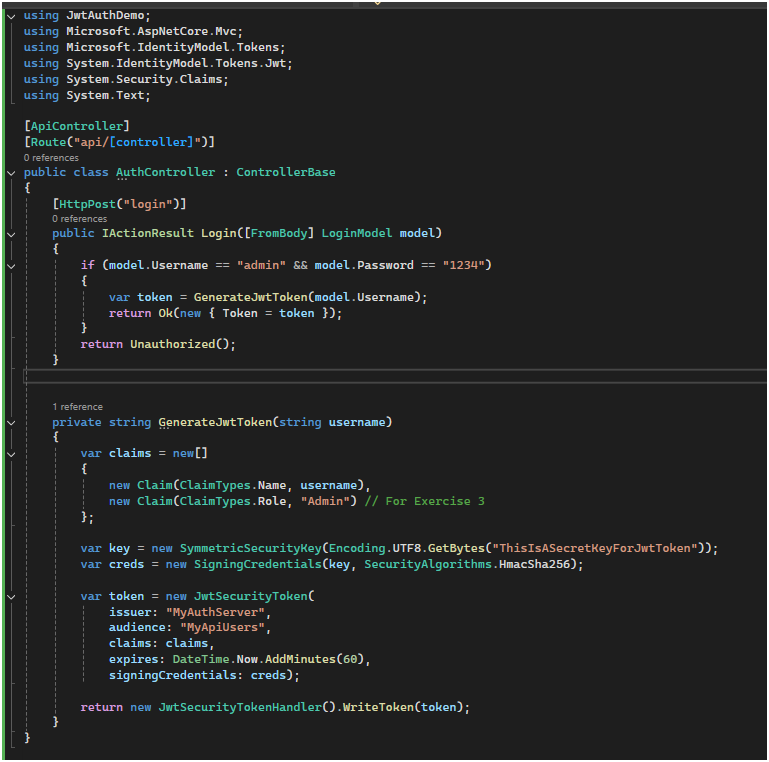
****

**4.Create a Models/LoginModel.cs file**

**5.Configure JWT in Program.cs**

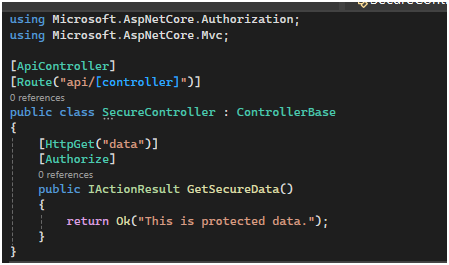
****

**6.Create Controllers/AuthController.cs**

****

**Exercise 2:** **Secure an API Endpoint**

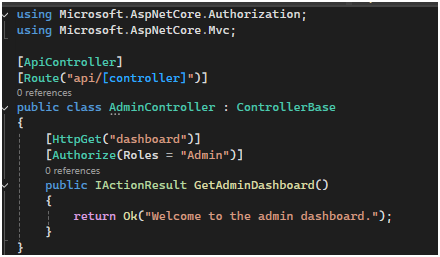
**1.Create Controllers/SecureController.cs**

****

**EXERCISE 3: Role-Based Authorization**

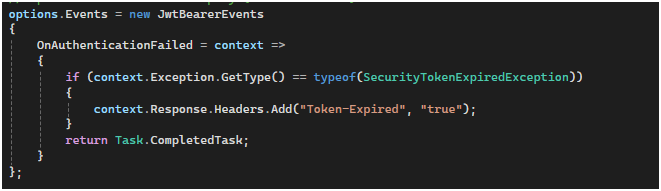
**1.Already added ClaimTypes.Role = "Admin" in token generation above.**

**2.Create Controllers/AdminController.cs**

****

**EXERCISE 4: Token Expiry Handling**

**Already added this part in Program.cs:**

****