**📦 Real-Life Scenario: E-Commerce Admin Panel with Role-Based API Access**

**🛒 Scenario: Manage Products in an E-Commerce System**

**Problem Statement:**

Build a ProductController in an ASP.NET Core Web API secured using JWT token-based authentication.

**✅ Requirements:**

1. **Customer Role (CUSTOMER)**
   * Can view the list of all products (GET /api/Product)
   * Can view a specific product (GET /api/Product/{id})
2. **Seller Role (SELLER)**
   * Can **add** a new product (POST /api/Product)
   * Can **update** their own products (PUT /api/Product/{id}) — only if the product belongs to them (check by SellerId)
3. **Admin Role (ADMIN)**
   * Can **delete** any product (DELETE /api/Product/{id})
   * Can view all products and sellers
4. All endpoints should be secured with [Authorize] and role-based [Authorize(Roles = "...")] where needed.
5. When a product is added or updated, attach the SellerId from the JWT token (don’t accept it from the body to prevent spoofing).