Product.cs

namespace Domain

{

public class Product

{

public int Id { get; set; }

public string Name { get; set; }

public decimal Price { get; set; }

public int CategoryId { get; set; }

}

}

Category.cs

namespace Domain

{

public class Category

{

public int Id { get; set; }

public string Name { get; set; }

}

}

Appdbcontext.cs

using Microsoft.EntityFrameworkCore;

namespace Data

{

public class AppDbContext : DbContext

{

public DbSet<Product> Products { get; set; }

public DbSet<Category> Categories { get; set; }

public AppDbContext(DbContextOptions<AppDbContext> options) : base(options)

{

}

}

}

ProductService.cs

using System.Collections.Generic;

using System.Linq;

namespace Services

{

public class ProductService

{

private readonly AppDbContext \_context;

public ProductService(AppDbContext context)

{

\_context = context;

}

public IEnumerable<Product> GetAllProducts()

{

return \_context.Products.ToList();

}

// Implement other CRUD methods

}

}

Appsettings.cs

{

"ConnectionStrings": {

"DefaultConnection": "your\_database\_connection\_string"

}}

Startup.cs

using Microsoft.EntityFrameworkCore;

using Microsoft.Extensions.Configuration;

using Microsoft.Extensions.DependencyInjection;

namespace WebApiApp

{

public class Startup

{

public IConfiguration Configuration { get; }

public Startup(IConfiguration configuration)

{

Configuration = configuration;

}

public void ConfigureServices(IServiceCollection services)

{

// Other configurations

services.AddDbContext<AppDbContext>(options =>

options.UseSqlServer(Configuration.GetConnectionString("DefaultConnection")));

services.AddScoped<ProductService>();

}

}

Controllers.cs

using Microsoft.AspNetCore.Mvc;

using Services;

namespace WebApiApp.Controllers

{

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

[ApiController]

public class ProductsController : ControllerBase

{

private readonly ProductService \_productService;

public ProductsController(ProductService productService)

{

\_productService = productService;

}

[HttpGet]

public ActionResult<IEnumerable<Product>> GetProducts()

{

var products = \_productService.GetAllProducts();

return Ok(products);

}

}

}