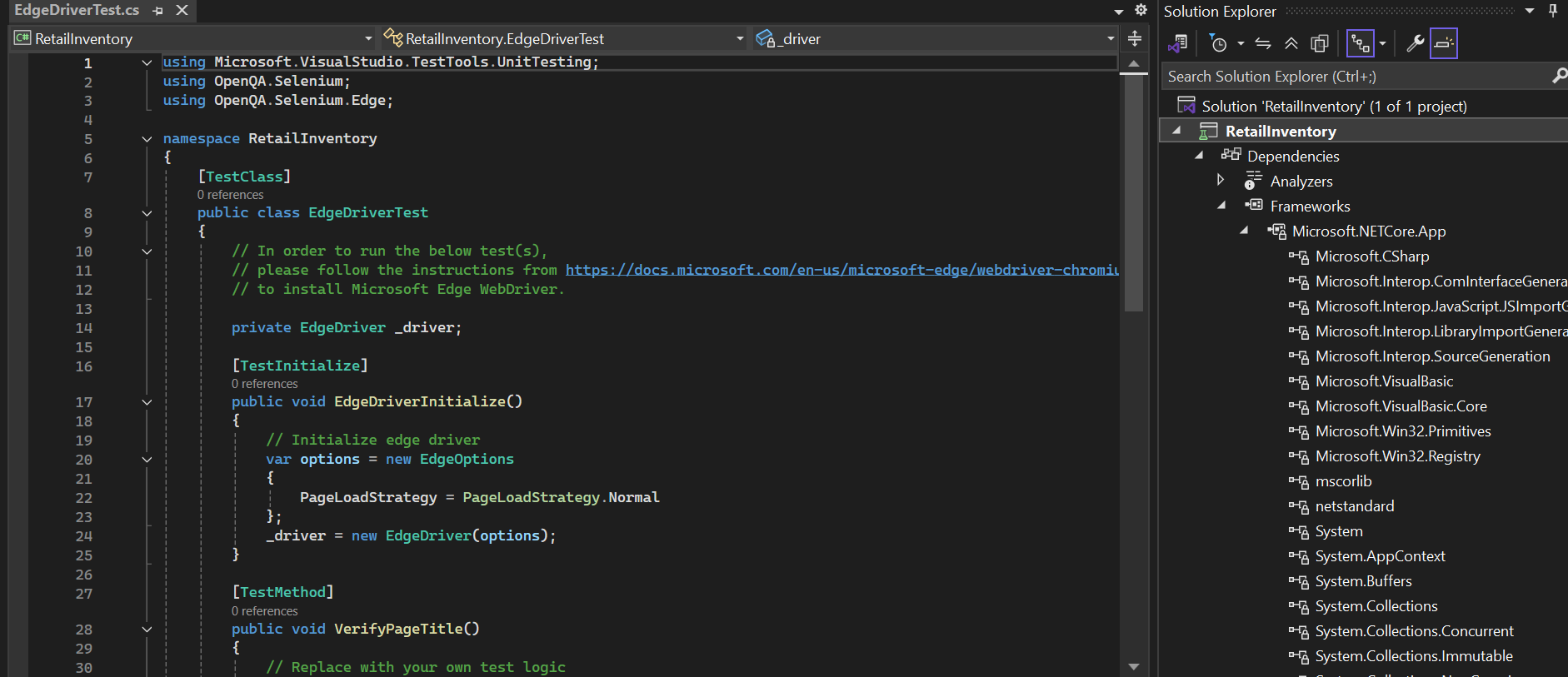
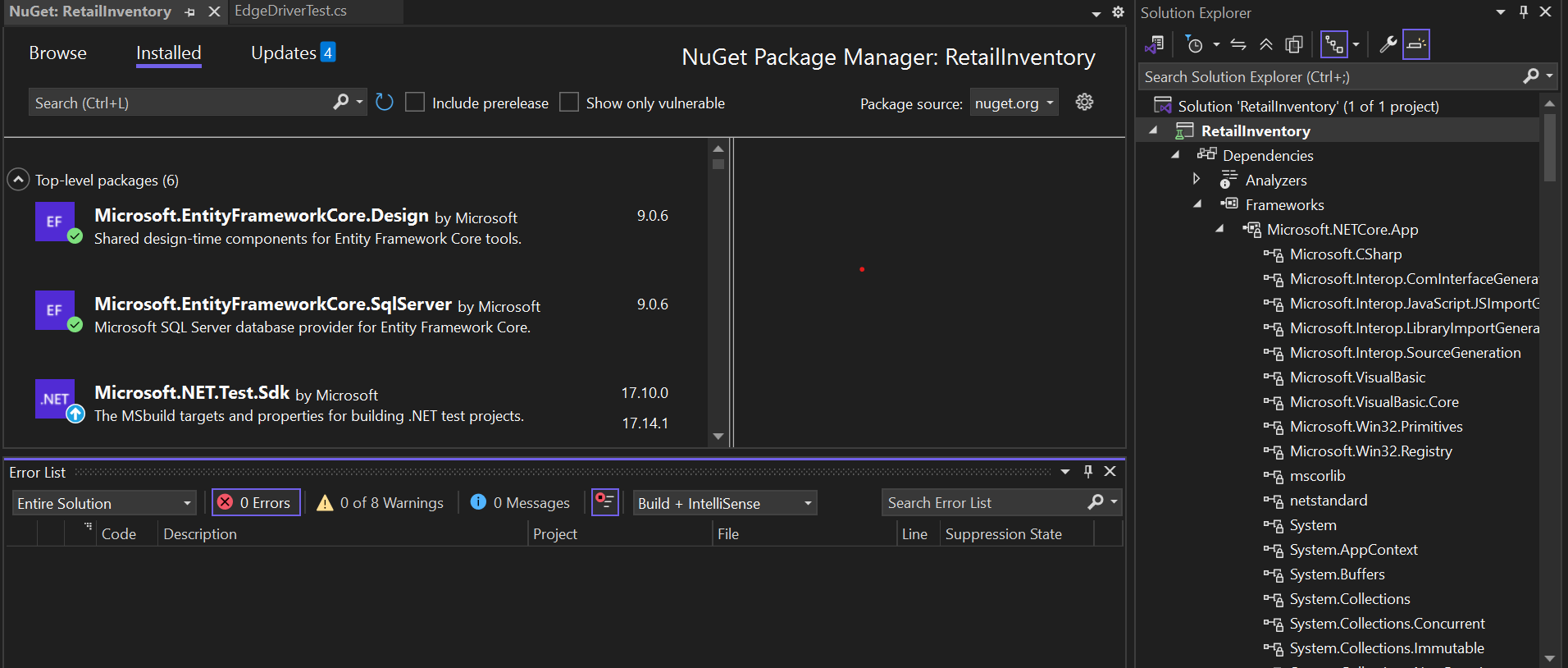
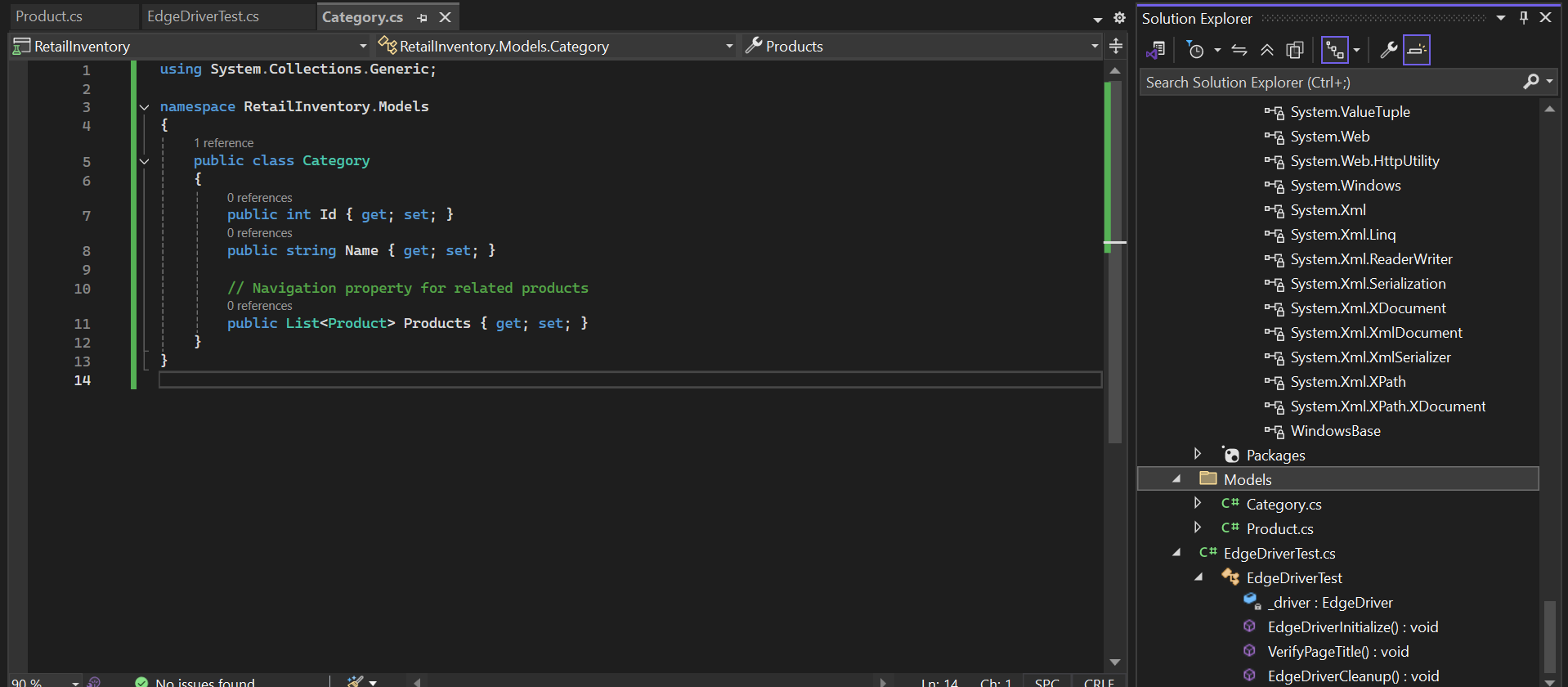
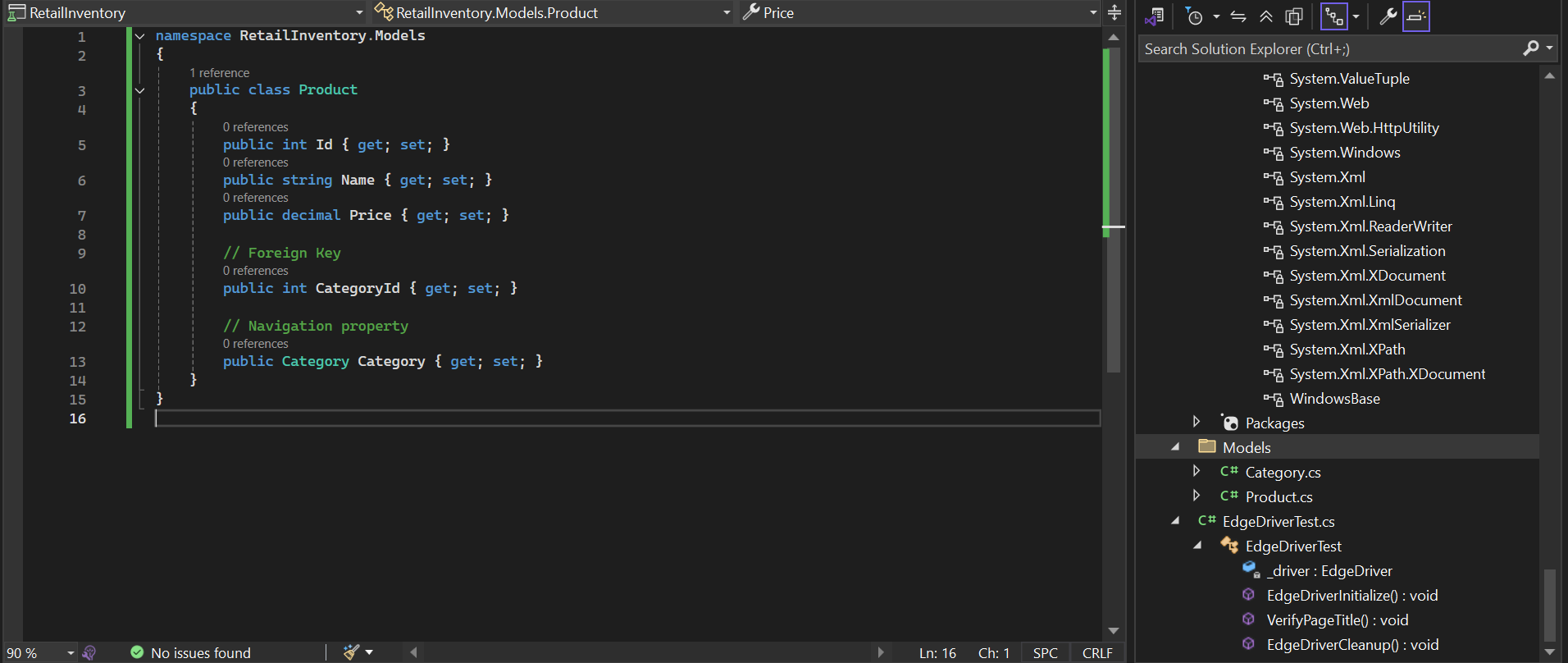
**Lab 1: Understanding ORM with a Retail Inventory System**



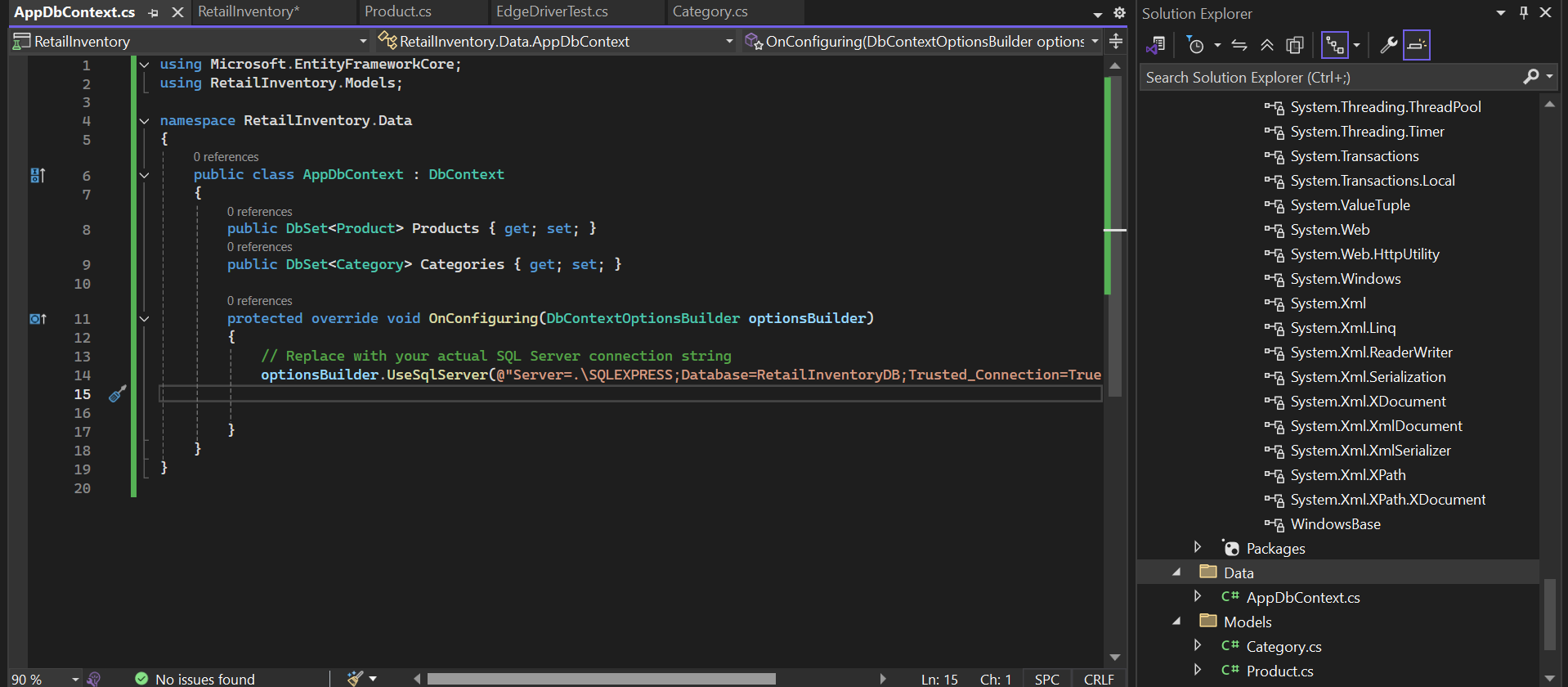


**Lab 2: Setting Up the Database Context for a Retail Store**

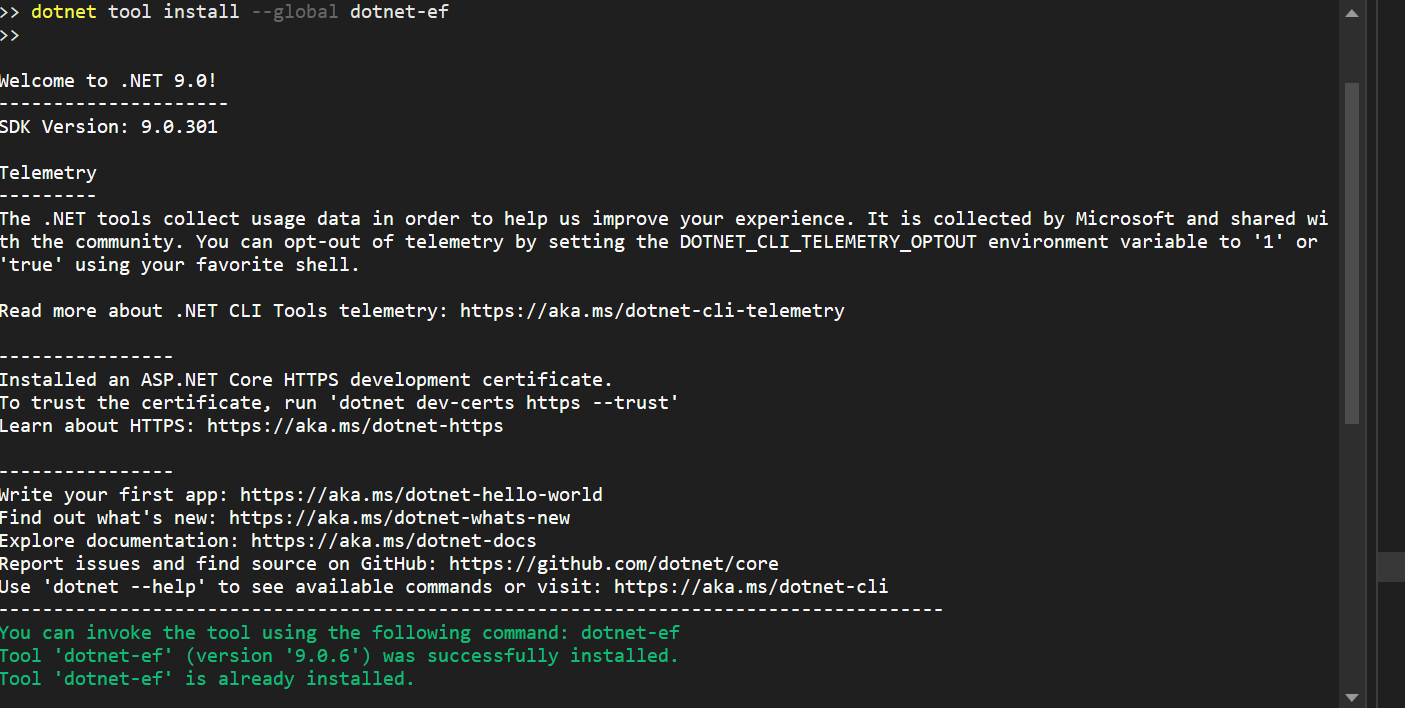




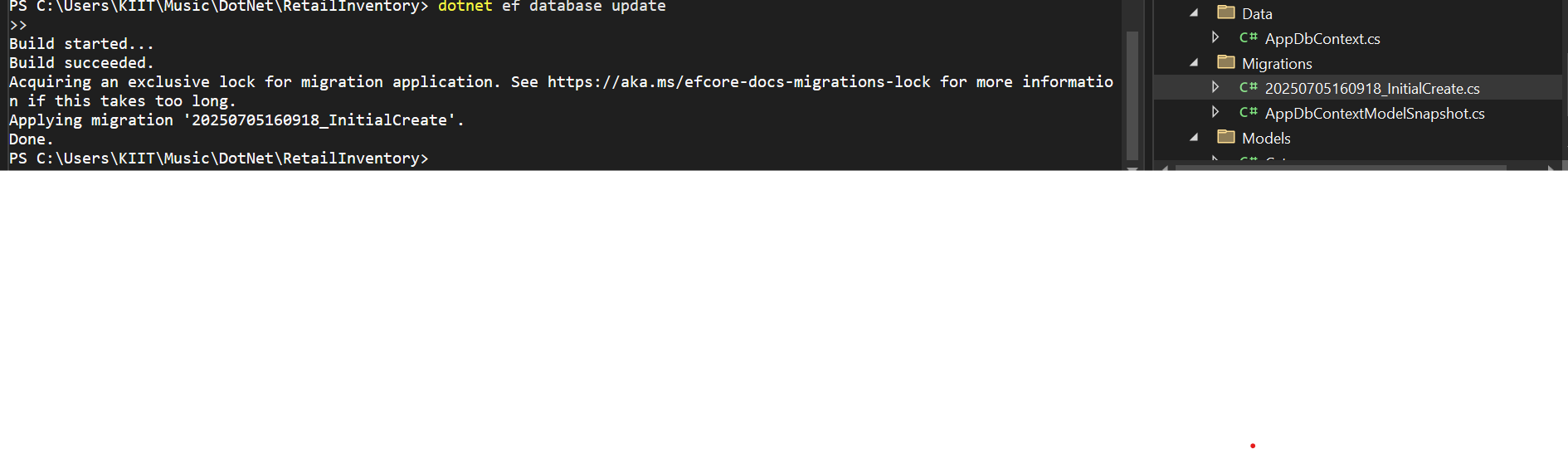
**AppDbContext.cs**



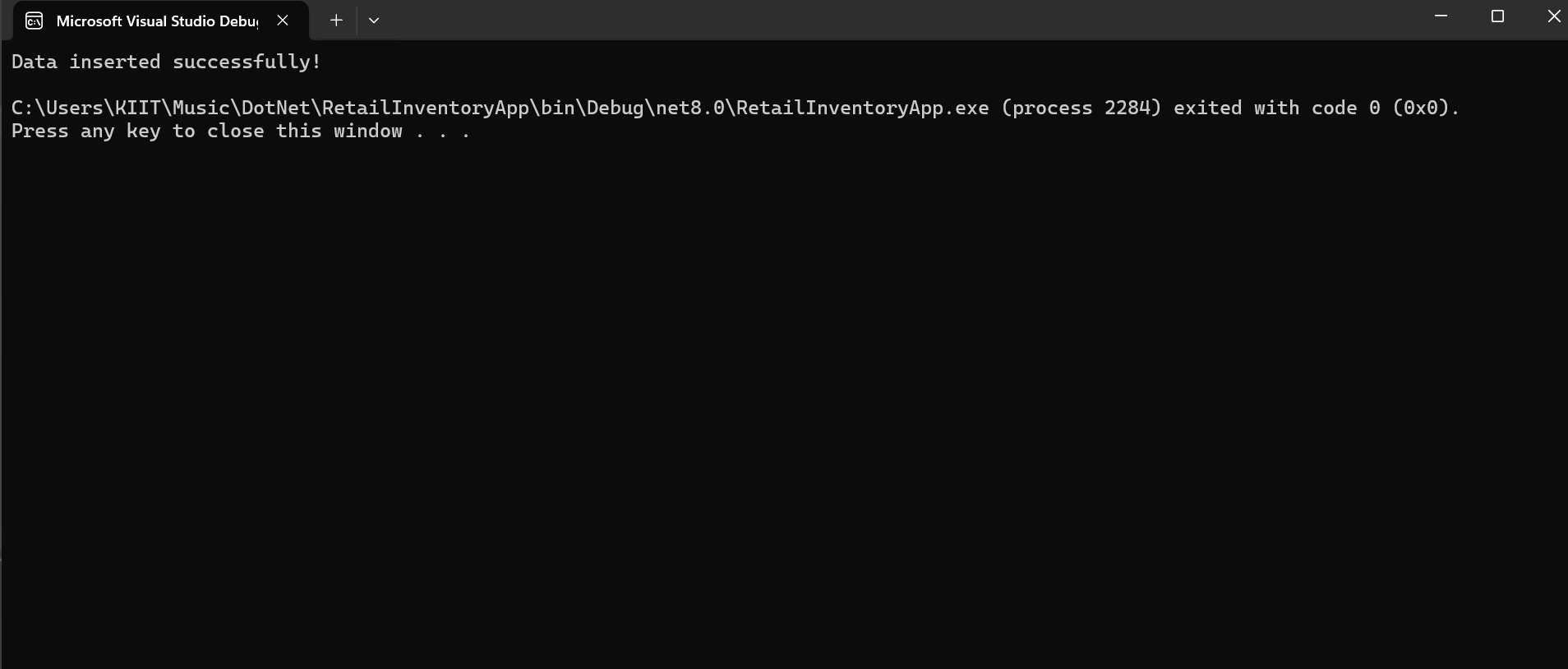
**Lab 3: Using EF Core CLI to Create and Apply Migrations**

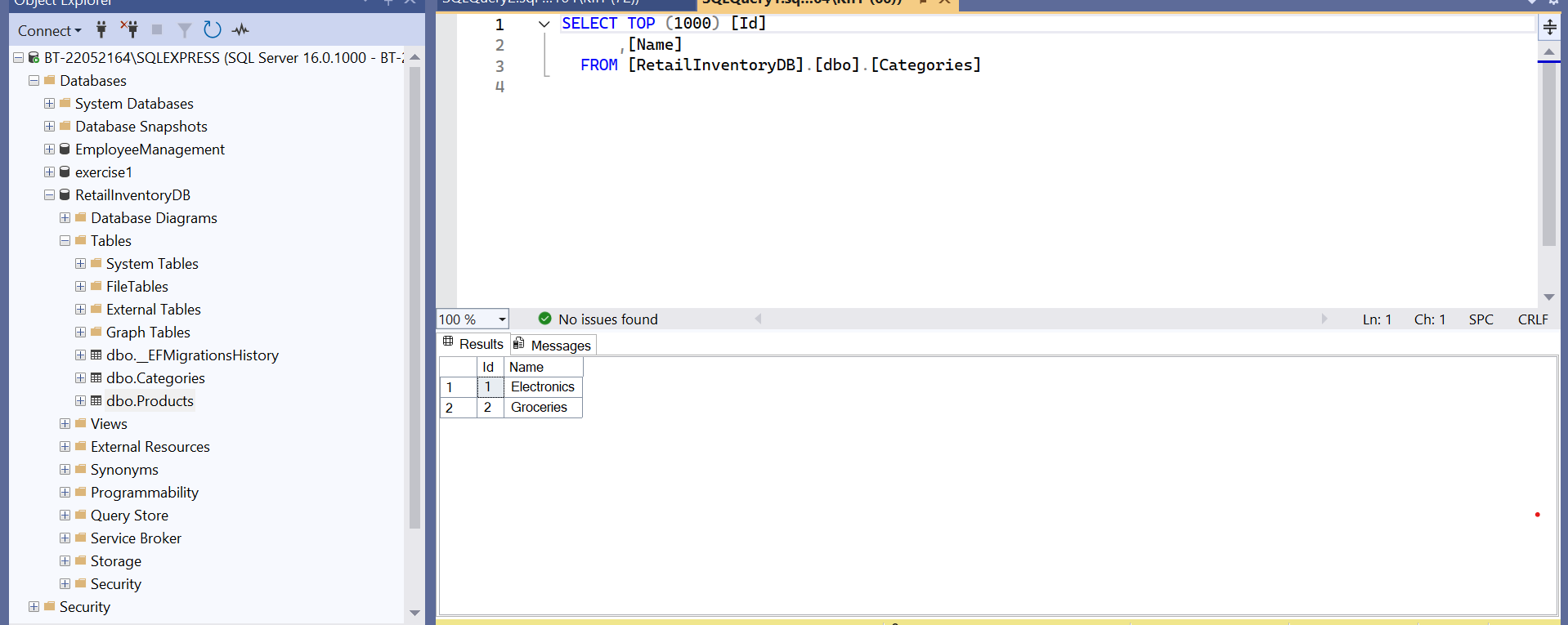


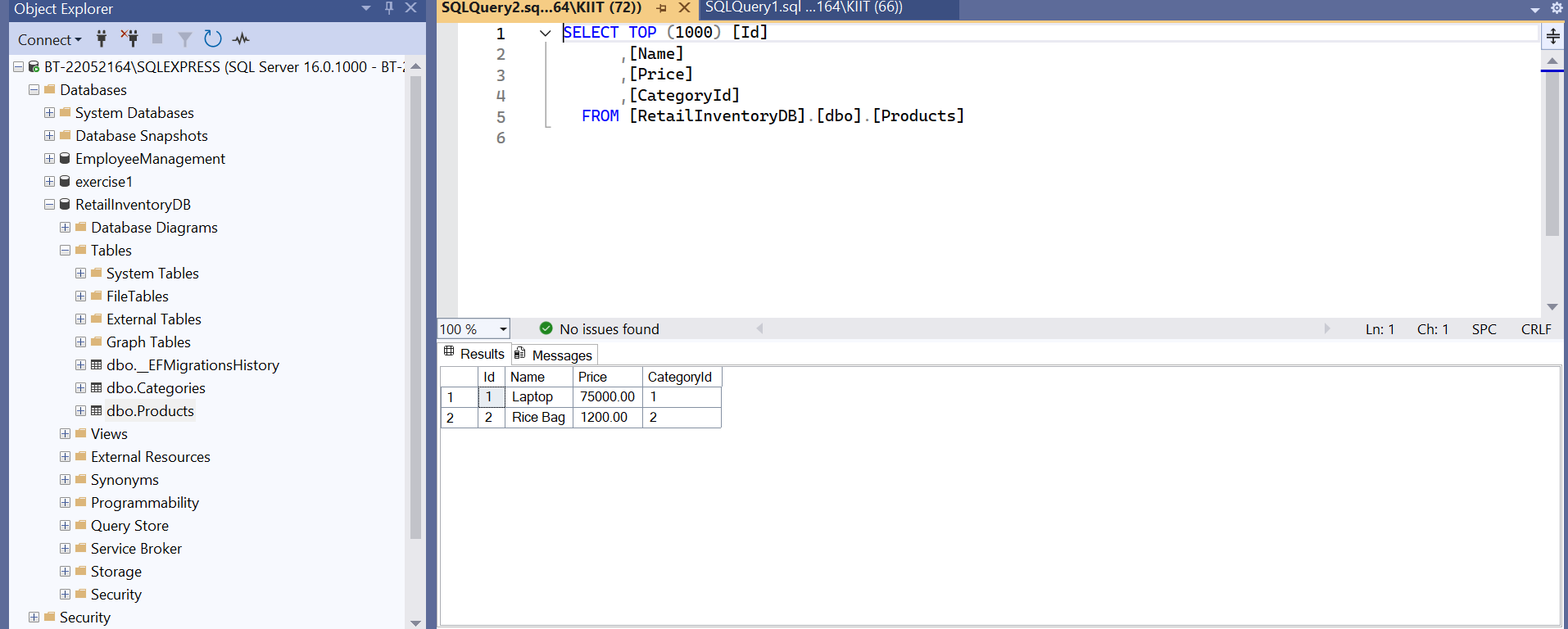


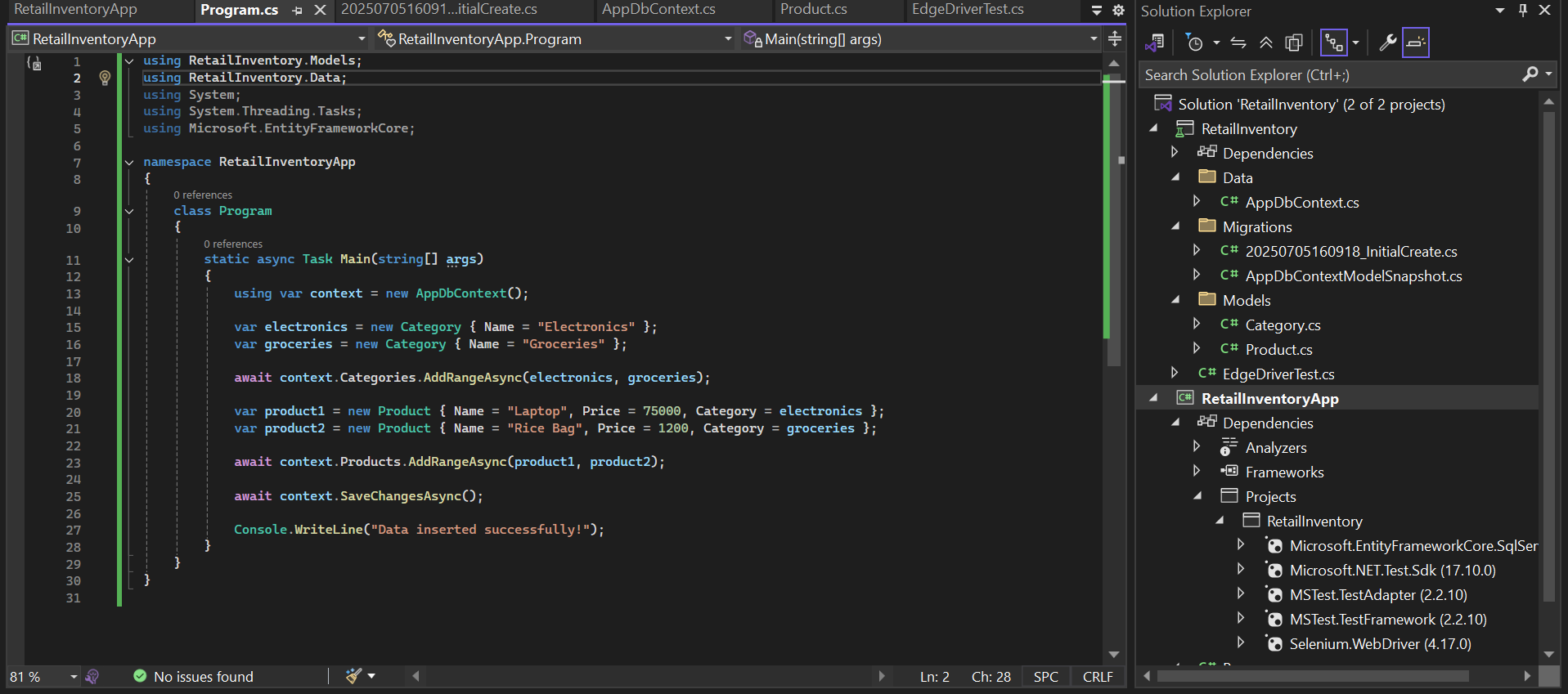
**DATABASE Successfully created**

**Lab 4: Inserting Initial Data into the Database**









**Lab 5: Retrieving Data from the Database**

using RetailInventory.Models;

using Microsoft.EntityFrameworkCore;

using RetailInventory.Data;

using System;

using System.Threading.Tasks;

namespace RetailInventoryApp

{

class Program

{

static async Task Main(string[] args)

{

using var context = new AppDbContext();

Console.WriteLine("Retrieving all products from the database...\n");

var products = await context.Products.ToListAsync();

foreach (var p in products)

Console.WriteLine($"{p.Name} - ₹{p.Price}");

Console.WriteLine("\nFinding product by ID...\n");

var product = await context.Products.FindAsync(1);

if (product != null)

Console.WriteLine($"Found: {product.Name}");

else

Console.WriteLine("Product not found");

Console.WriteLine("\nFinding first expensive product (price > ₹50000)...\n");

var expensive = await context.Products.FirstOrDefaultAsync(p => p.Price > 50000);

if (expensive != null)

Console.WriteLine($"Expensive product: {expensive.Name}");

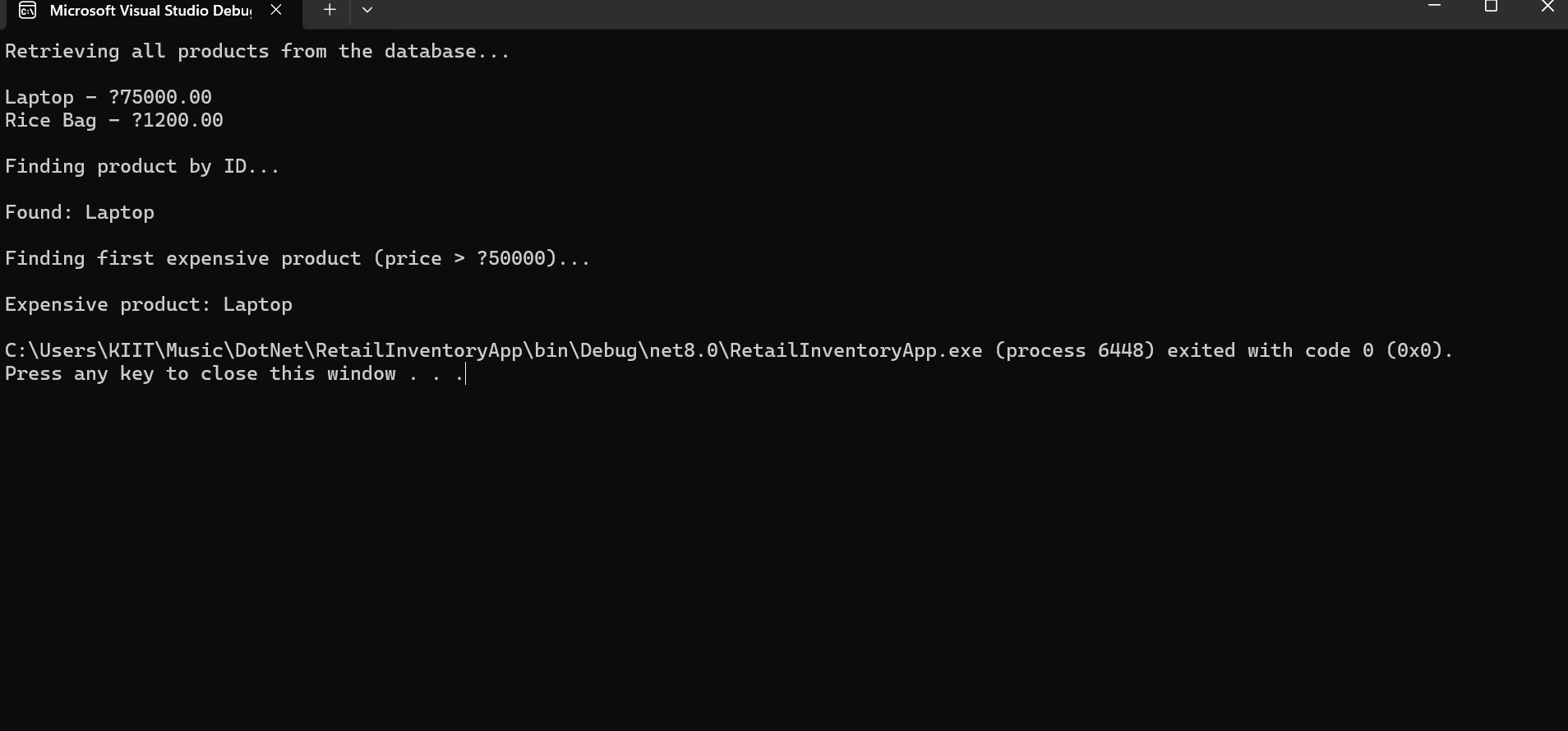
else

Console.WriteLine("No expensive product found");

}

}

}

****