# WEEK- 3 ENTITY FRAMEWORK CORE

**LAB -1 UNDERSTANDING ORM WITH A RETAIL INVENTORY SYSTEM**

**What is ORM?**

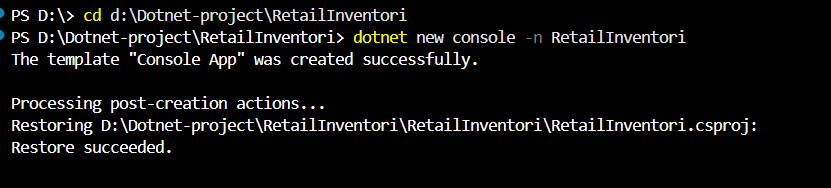
Object-Relational Mapping (ORM) is a programming technique that allows developers to interact with a database using object-oriented programming (OOP) concepts instead of writing raw SQL queries. It maps C# classes (models) to database tables, and class properties to table columns.

* **Class - Table**: A C# class (e.g., Product) represents a database table (Products).
* **Properties - Columns**: Each property (e.g., Id, Name) maps to a column.
* **Relationships - Foreign Keys**: Associations (e.g., Product.Category) map to foreign keys.

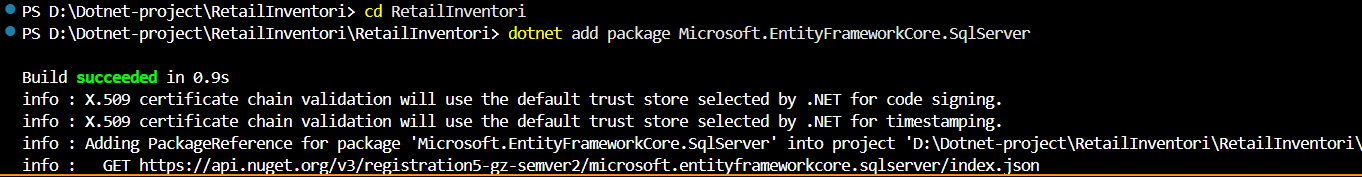
### **Benefits of ORM**

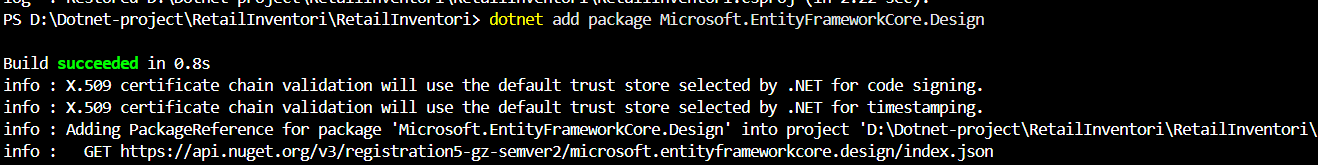
**Productivity** – Write C# code instead of SQL (e.g., db.Products.Add(product)).  
**Maintainability** – Automatic schema updates via **migrations**.  
**Database Agnostic** – Works with SQL Server, PostgreSQL, MySQL, etc.  
**Security** – Reduces SQL injection risks by using parameterized queries.

Creating a new project



Installing EF Core packages

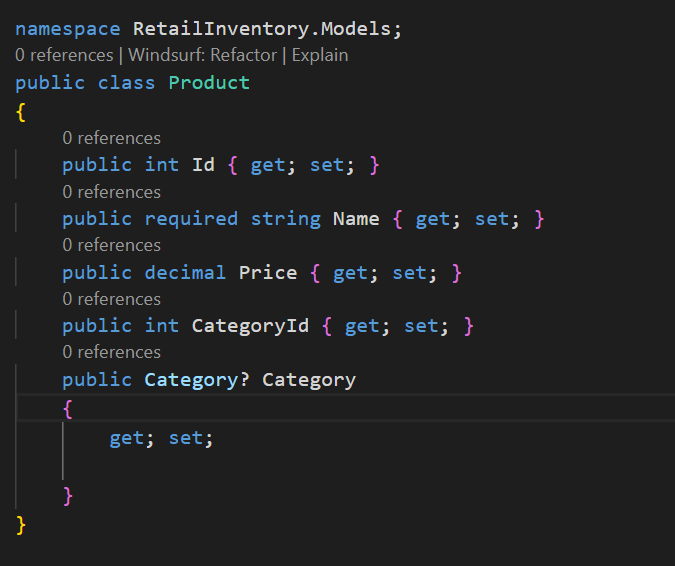




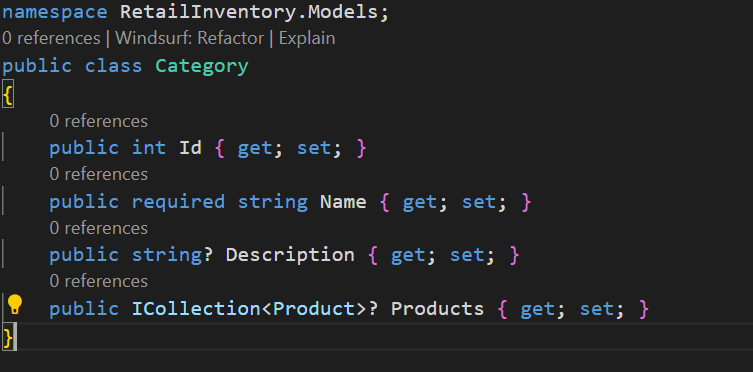
**LAB 2: SETTING UP THE DATABASE CONTEXT FOR A RETAIL STORE**

**1)**

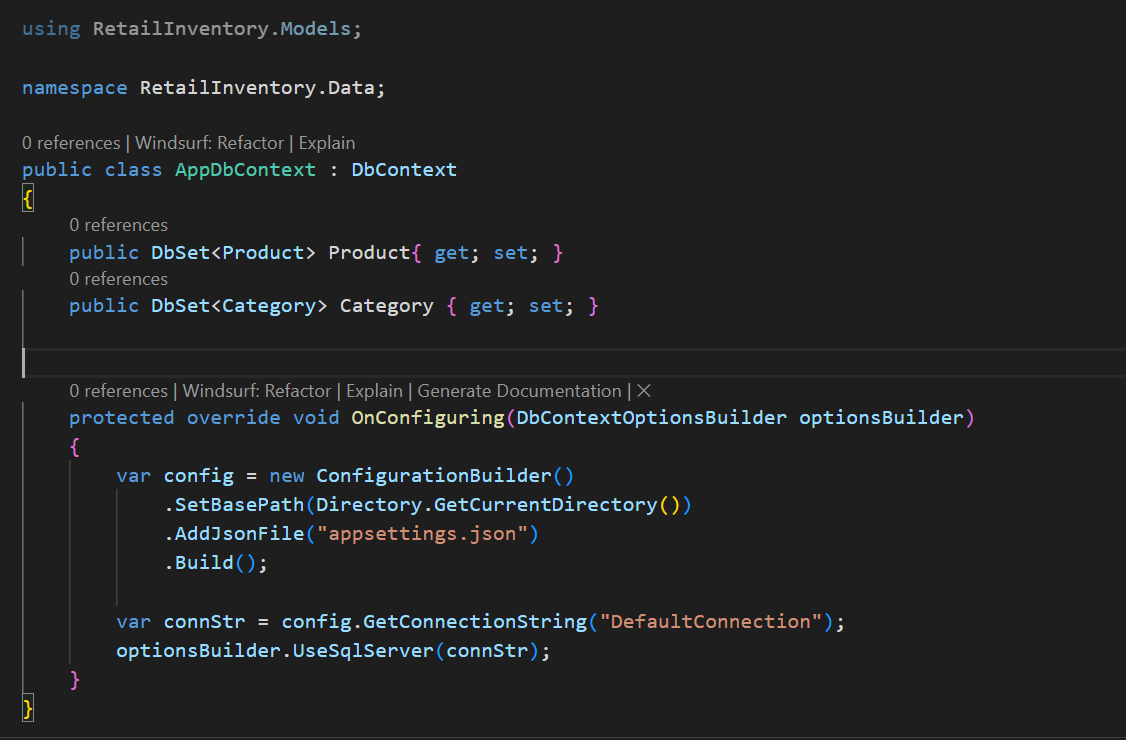
**Product.cs**

****

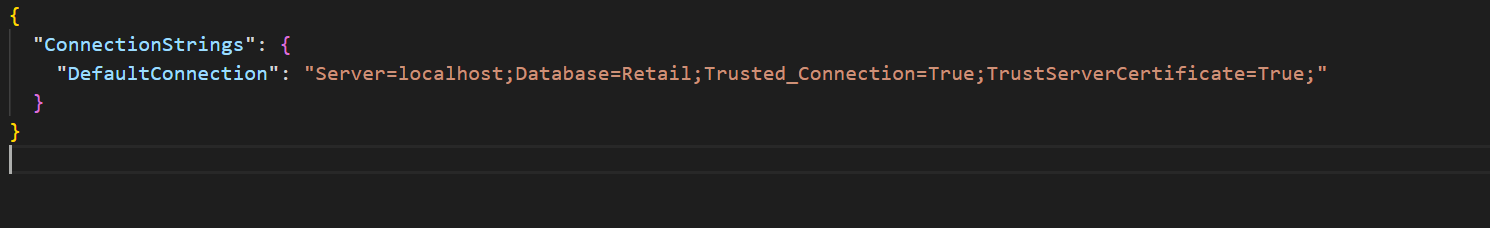
**Category.cs**

****

2) AppDbContext

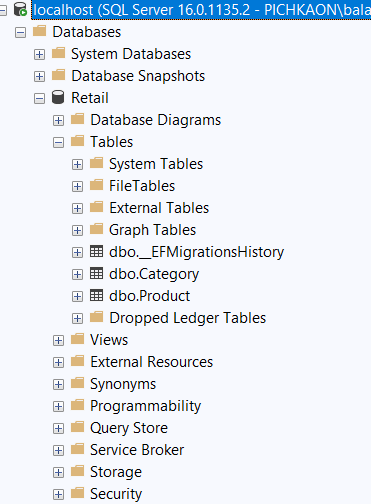


3) appsettings.json



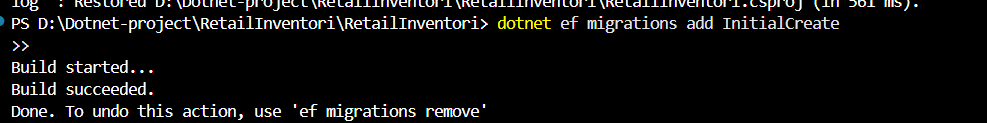
Connection String : **@"Server=localhost;Database=Retail;Trusted\_Connection=True;TrustServerCertificate=True;"**

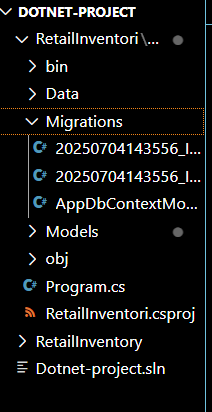
Database in SSMS



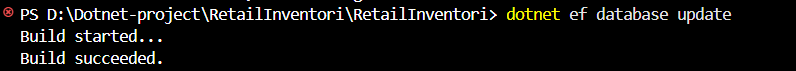
**LAB 3: USING EF CORE CLI TO CREATE AND APPLY MIGRATIONS**

Create Initial Migration

****

****

Apply Migration to Create Database:

****

Updation of database

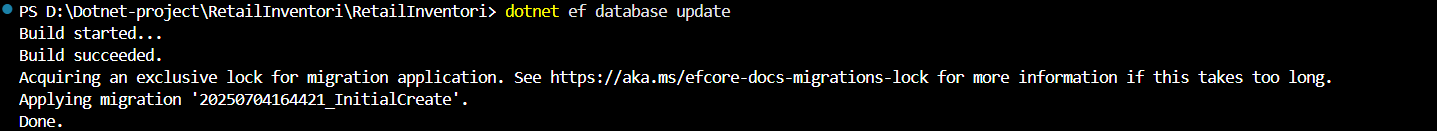
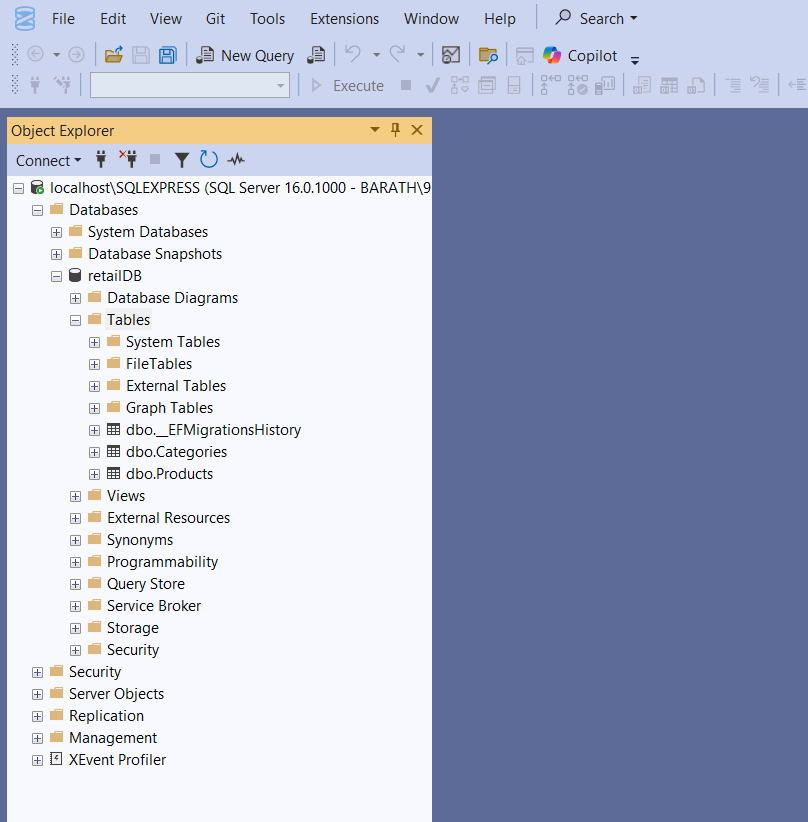
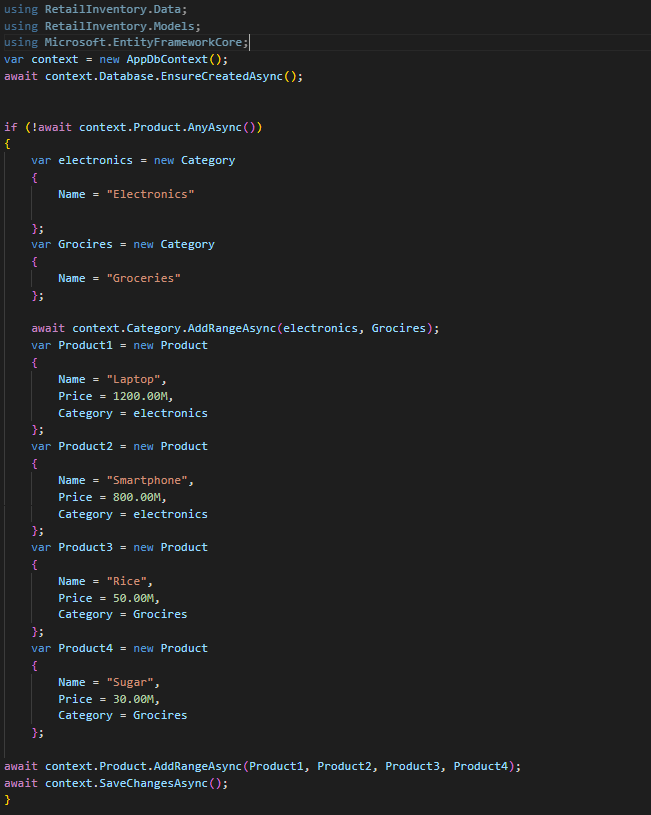
****

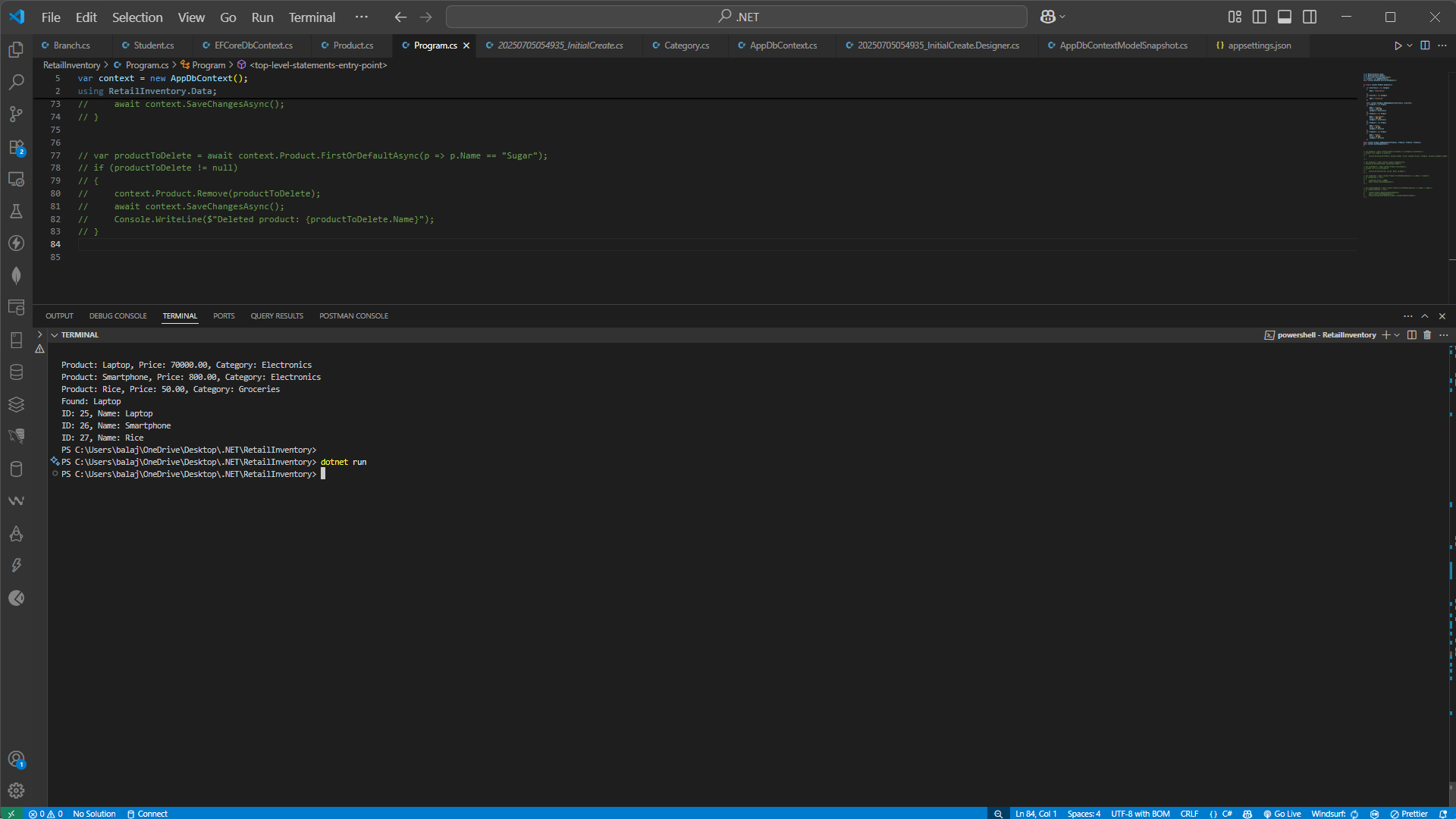
Table Column in SSMS

****

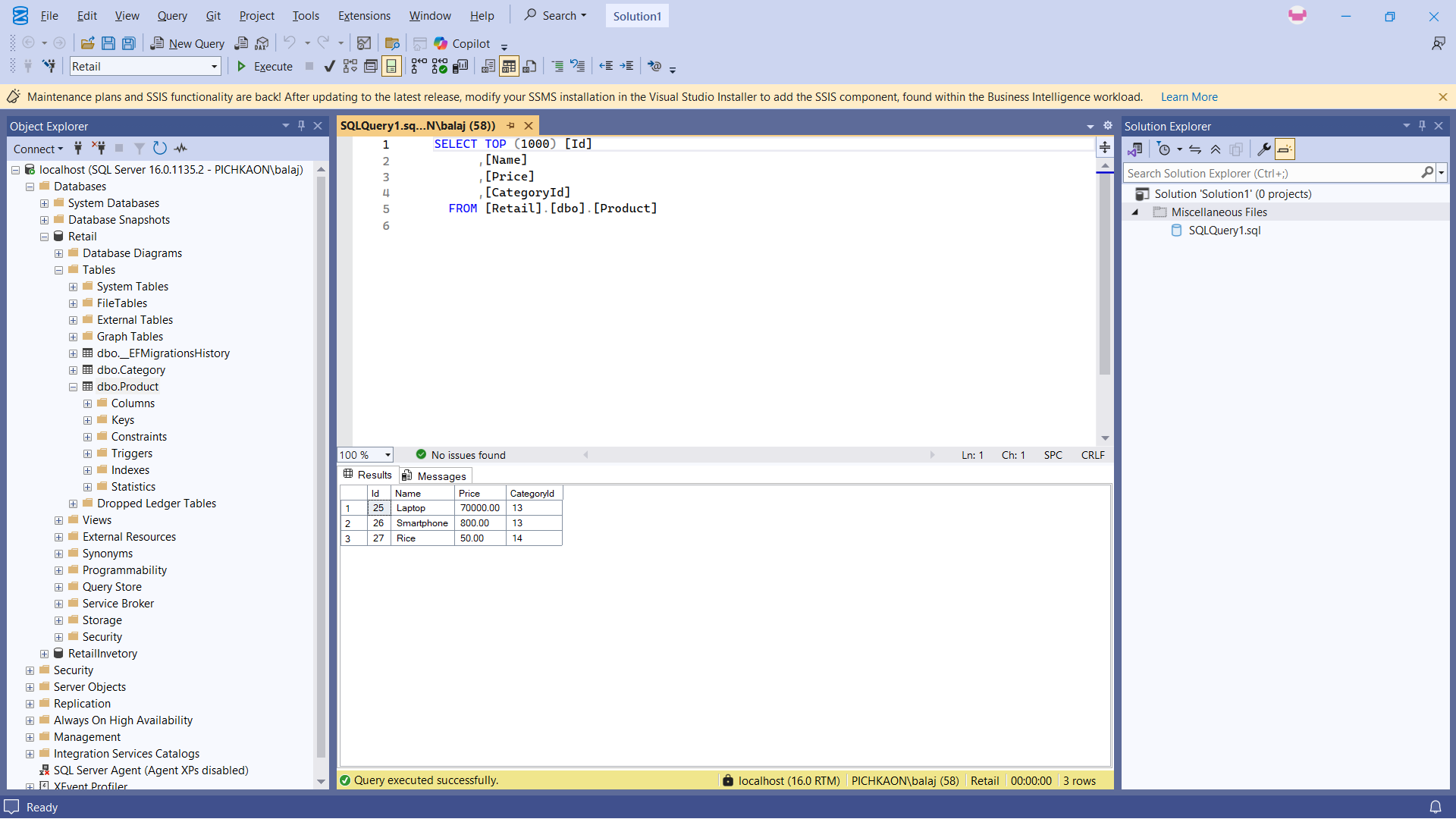
**LAB 4: INSERTING INITIAL DATA INTO THE DATABASE**

**Insertng items in product table in Program.cs**

**Running the application**

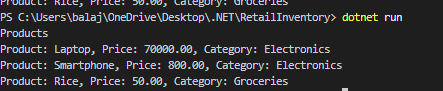


**Using sql queryfor checking table updation**

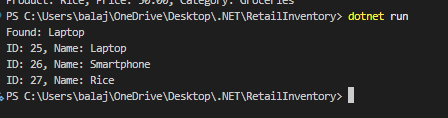


**LAB 5: RETRIEVING DATA FROM THE DATABASE**

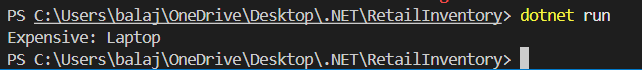
1. **Retrieve All Products:**

****

1. **Find by ID**

****

1. **FirstOrDefault with Condition:**

****