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

1. What is ORM?

ORM (Object-Relational Mapping) is a technique used in software development to map programming language objects (like C# classes) to database tables. It allows you to interact with a database using C# code instead of writing raw SQL queries.

Example Mapping:

| **C# Class** | **SQL Table** |
| --- | --- |
| Product | Products |
| Name property | Name column |

The ORM framework handles the conversion between your object-oriented code and the relational database structure behind the scenes.

Benefits of ORM:

Productivity: Developers can focus on writing C# code without worrying about SQL queries.

Maintainability: Centralized models make the codebase easier to update and manage.

Abstraction: Hides SQL logic behind C# methods like .Add(), .Remove(), or .SaveChanges().

1. EF Core vs EF Framework (EF6)

| **Feature** | **EF Core** | **EF Framework (EF6)** |
| --- | --- | --- |
| Platform | Cross-platform (.NET Core/.NET 5+) | Windows-only |
| Lightweight | Yes | No (heavier) |
| Performance | Faster due to modern optimizations | Slower in comparison |
| LINQ & Async Support | Full support | Partial/less efficient |
| Compiled Queries | Supported (faster performance) | Not available |
| Use Case | Recommended for new development | Legacy support only |

3. EF Core 8.0 Features

EF Core 8.0 brings several enhancements for better performance and modern use cases:

JSON Column Mapping: You can store and retrieve entire objects as JSON in a single SQL column.

Compiled Models: Reduces app startup time by avoiding model re-parsing.

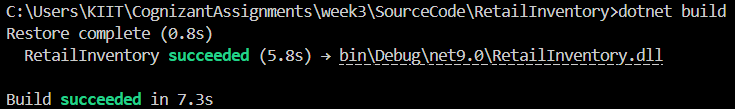
Interceptors: Lets you hook into database commands for logging, auditing, etc.

Bulk Operations: Better handling of large inserts/updates in batches.

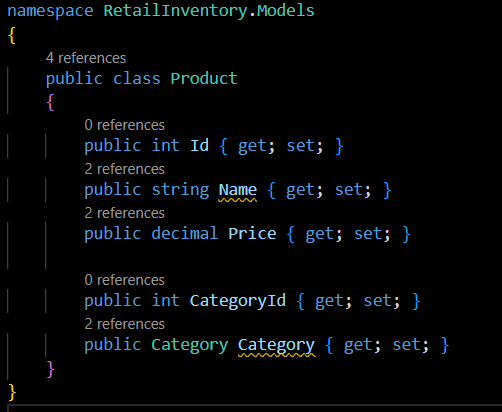
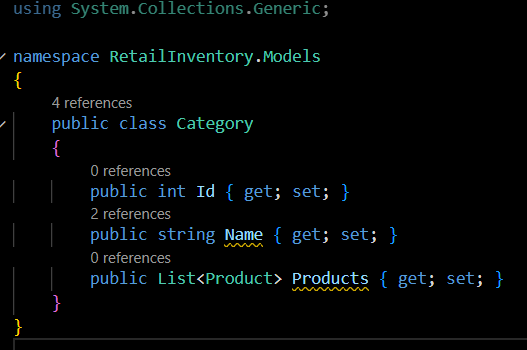
4. Create a .NET Console App:



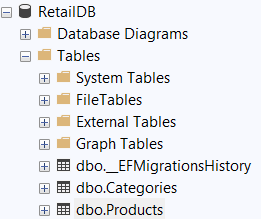
1. Install EF Core Packages:

Screenshot 2025-07-05 224951Screenshot 2025-07-05 225009

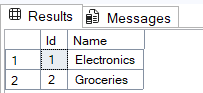
**Lab 2: Setting Up the Database Context for a retail Store**



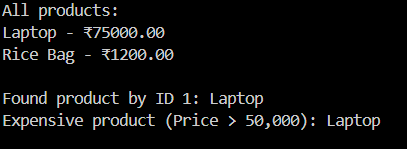
**Lab 3: Using EF Core CLI to Create and Apply Migrations**

****

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

****

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

****