**Superset-Id: 6364597  
Name: SRINJOY PAUL**

**WEEK- 3 ENTITY FRAMEWORK CORE 8.0 HANDSON**

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

**Step-1) What is ORM ? Explain how ORM maps C# classes to database tables? What are the benefits of ORM?  
Answer:**

ORM is a development technique that enables coders to work with databases using language-specific objects instead of writing direct SQL queries**.**  -Each class in C# is equivalent to a database table.

-The attributes (properties) of the class represent the table's fields.

-A class object symbolizes a single record in the database.

-**Productivity**: Minimizes repetitive SQL code.  
-**Maintainability:** Code-first methodology makes updates simpler.

- **Abstraction:** LINQ queries replace manual SQL writing**.**

**Step-2) Difference between EF Core and EF Framework**

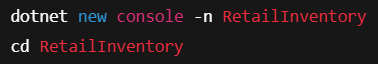
**Answer:  
  
Feature: Platform:-  
EF Core:** Compatible with multiple platforms (.NET Core)  
**EF Framework:** Limited to Windows (.NET Framework)  
 **Feature:Speed:-   
EF Core:**  Leaner and offers better speed  
**EF Framework:** Stable but bulkier  
  
**Feature:Capabilities:-**

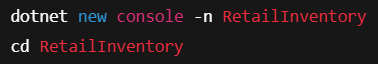
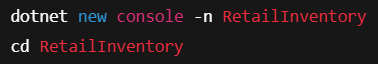
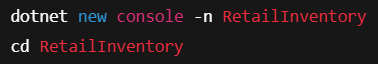
**EF Core:** Offers modern features like async, LINQ, compiled models  
**EF Framework:** Lacks several updated functionalities  
  
**Feature:Best Fit For:-  
EF Core:** New apps, especially for cloud or cross-platform **EF Framework:** Old systems, typically for enterprises

**Step-3) What are the new features of EF Core 8.0?  
Answer:  
JSON Support:** Allows use of JSON fields in SQL Server.

**Precompiled Models:** Helps boost app launch time and runtime performance.

**Interceptors: Lets you tap into DB actions for custom code.**

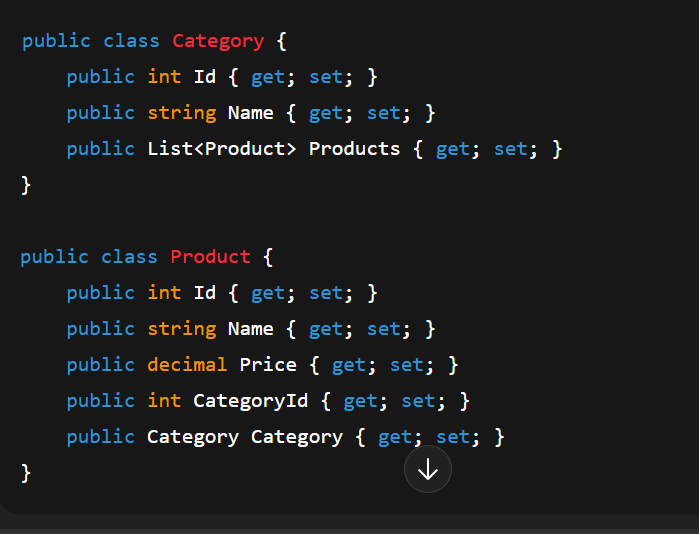
**Mass Updates:** Enhanced handling of bulk operations like insert/update/delete**.  
  
Step-4) Create a .NET Console App  
Answer:**

**Step-5) Install EF Core Packages:**

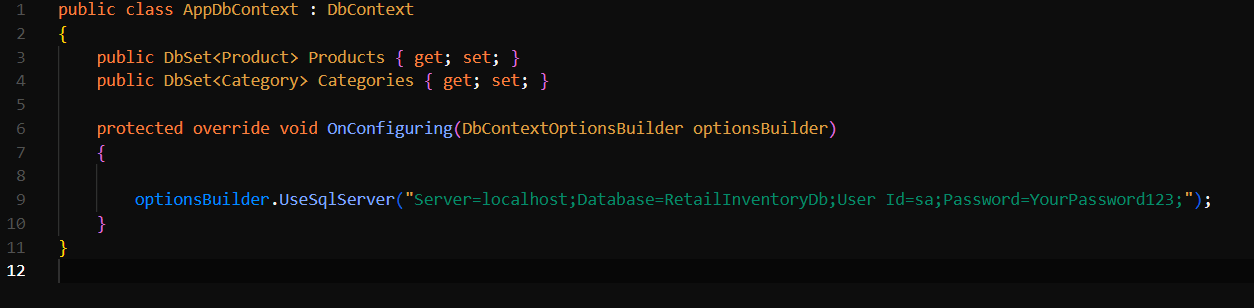
**Answers:**

**Picture2**

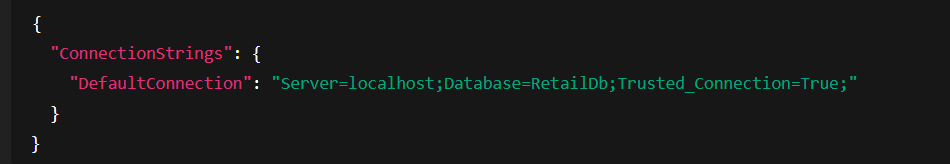
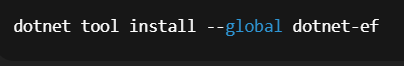
**Lab 2: Setting Up the Database Context for a Retail Store**

**Step-1) Create Models:  
**

**Step- 2. Create AppDbContext:**

****

**Step-3. Add Connection String in appsettings.json (optional for ASP.NET Core).**

  
  
  
  
  
**Lab 3: Using EF Core CLI to Create and Apply Migrations  
  
Step-1 Install EF Core CLI**

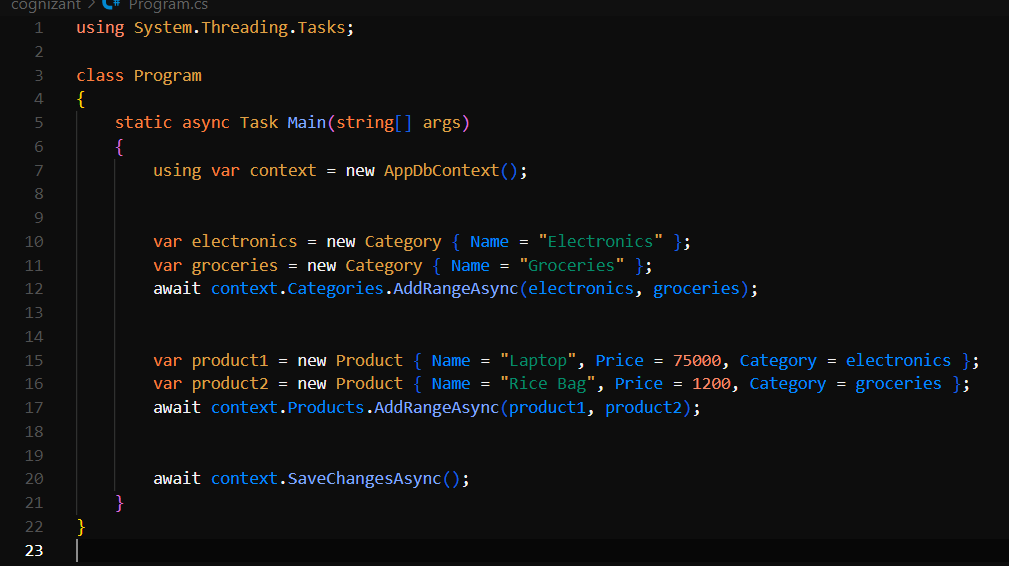
**Step-2:Create Initial Migration**

**Step-3: Apply Migration to Create Database:**

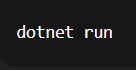
****

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

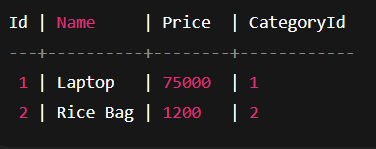
**Step-1. Insert Data in Program.cs:**

****

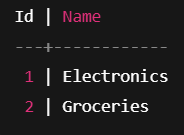
**Step- 2. Run the App:**

****

**Step-3: Verify in SQL Server:**

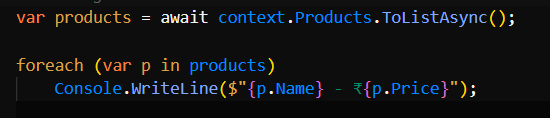
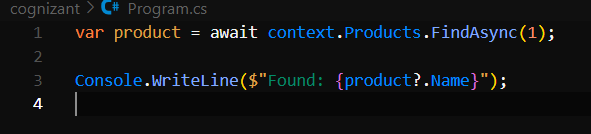
**Products Table:**

**Categories Table:**

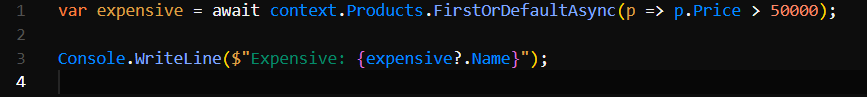


**Lab 5: Retrieving Data from the Database**

**Step-1. Retrieve All Products:**

**  
  
  
Step-2: Find by ID:  
  
**

**Step-3: FirstOrDefault with Condition:**

****