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

**Scenario:**

You’re building an inventory management system for a retail store. The store wants to

track products, categories, and stock levels in a SQL Server database.

**Objective:**

Understand what ORM is and how EF Core helps bridge the gap between C# objects and

relational tables.

**Steps:**

1. **What is ORM?**

• Explain how ORM maps C# classes to database tables.

• Benefits: Productivity, maintainability, and abstraction from SQL.

ORM (Object-Relational Mapping) is a programming technique that allows developers to work with relational databases by using object-oriented programming languages such as C#. It maps C# classes to SQL database tables and class properties to columns of the table.

Most important advantages of ORM:

It enhances productivity for developers by minimizing the amount of raw SQL queries needed to be written.

It makes it easier to maintain, since changes in the database schema can be reflected in code by using migrations.

It offers abstraction so that developers can work at a higher level without directly working with SQL.

2. **EF Core vs EF Framework:**

• EF Core is cross-platform, lightweight, and supports modern features like

LINQ, async queries, and compiled queries.

• EF Framework (EF6) is Windows-only and more mature but less flexible.

EF Core and EF Framework (EF6) are two iterations of Microsoft's library of ORM.

Entity Framework Core (EF Core):

It is lightweight and cross-platform.

It accommodates modern features like LINQ, async/await operations, and compiled models.

It is appropriate for new applications based on .NET 6/7/8.

Entity Framework (EF6):

It is Windows-only and bound to the older .NET Framework.

Although it is more mature, it has no support for most modern features.

It is best suited to legacy applications but less adaptable than EF Core.

3. **EF Core 8.0 Features:**

• JSON column mapping.

• Improved performance with compiled models.

• Interceptors and better bulk operations.

EF Core 8.0 brings a number of improvements that enhance performance and flexibility.

Key Features:

JSON column mapping: Supports storing and querying JSON directly in database columns.

Compiled models: Enhances startup performance.

Interceptors: Insert custom logic prior to or following EF Core operations.

Bulk operations: More optimized insert/update/delete of multiple records.

4. **Create a .NET Console App:**

dotnet new console -n RetailInventory

cd RetailInventory

**5. Install EF Core Packages:**

dotnet add package Microsoft.EntityFrameworkCore.SqlServer

dotnet add package Microsoft.EntityFrameworkCore.Design

