EE CORE 8.0 HANDS-ON

LAB-1

Question 1: What is ORM (Object-Relational Mapping)?

**ORM** is a technique that maps **objects in code** (like C# classes) to **tables in a relational database** (like SQL Server).

#### **Example Mapping:**

public class Product {

public int Id { get; set; }

public string Name { get; set; }

public int Stock { get; set; }

}

#### **Gets mapped to:**

|  |  |  |
| --- | --- | --- |
| **Id** | **Name** | **Stock** |
| 1 | "Laptop" | 5 |

#### **Benefits of ORM:**

* **Productivity:** Write less boilerplate SQL, focus on business logic.
* **Maintainability:** Schema and code evolve together via migrations.
* **Abstraction:** Write LINQ instead of raw SQL.

PTO

Question 2: EF Core vs EF Framework

|  |  |  |
| --- | --- | --- |
| **Feature** | **EF Core** | **EF Framework (EF6)** |
| Platform | Cross-platform (.NET Core/.NET 5+) | Windows-only (Full .NET Framework) |
| Performance | Lightweight, faster | Heavier but more mature |
| LINQ Support | Yes | Yes |
| Async Queries | Yes | Limited support |
| JSON Columns (v8.0+) | Yes | No |
| Compiled Queries | Yes | No |

Question 3: EF Core 8.0 – Key Features

* **JSON Column Mapping:** Store structured data (e.g., specs, metadata) directly in JSON SQL columns.
* **Compiled Models & Queries:** Improve startup and query execution performance.
* **Interceptors & Bulk Operations:** Better control over low-level events and efficient large-scale operations.

Question 4: Creating a .NET Console App and Installing EF Core Packages

