**WEEK 3 HANDS ON**

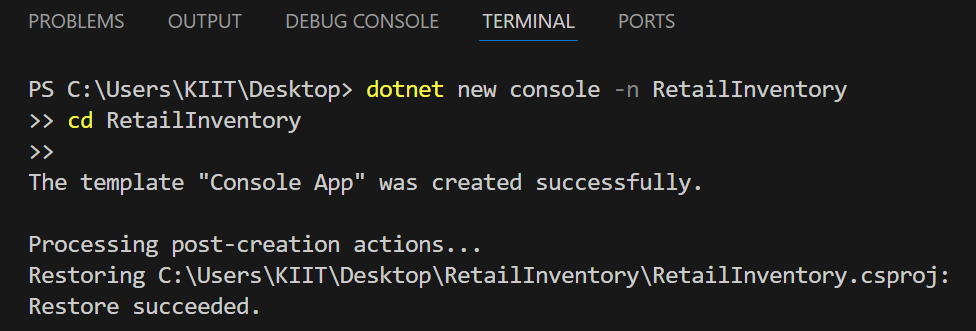
Submitted by:Simran Raghav(6362006)

**ENTITY FRAMEWORK CORE 8.0**

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

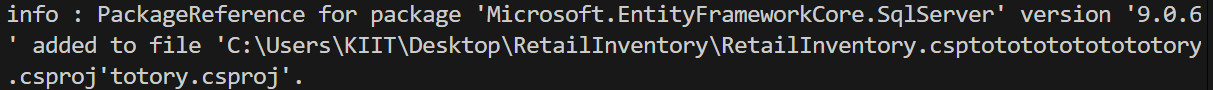
CODE:

**# creating the project**



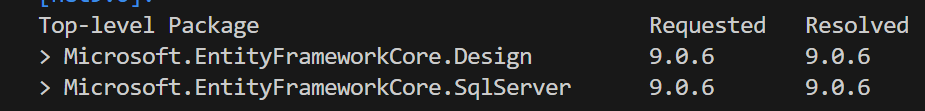
**# adding EF Core packages**

dotnet add package Microsoft.EntityFrameworkCore.SqlServer



dotnet add package Microsoft.EntityFrameworkCore.Design

dotnet list package



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

CODE:

**# Product.cs**

public class Product

{

    public int Id { get; set; }

    public string Name { get; set; }

    public decimal Price { get; set; }

    public int CategoryId { get; set; }

  public Category Category { get; set; }

}

**#Category.cs**

using System.Collections.Generic;

public class Category

{

    public int Id { get; set; }

    public string Name { get; set; }

    public List<Product> Products { get; set; }

}

**#AppDbContext.cs**

using Microsoft.EntityFrameworkCore;

public class AppDbContext : DbContext

{

    public DbSet<Product> Products { get; set; }

    public DbSet<Category> Categories { get; set; }

    protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)

    {

        optionsBuilder.UseSqlServer(

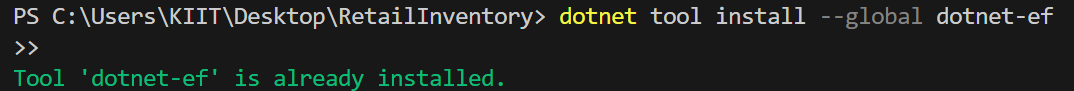
            "Server=.;Database=RetailDb;Trusted\_Connection=True;TrustServerCertificate=True;");

    }

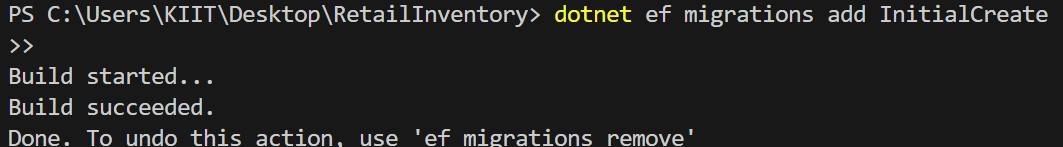
}

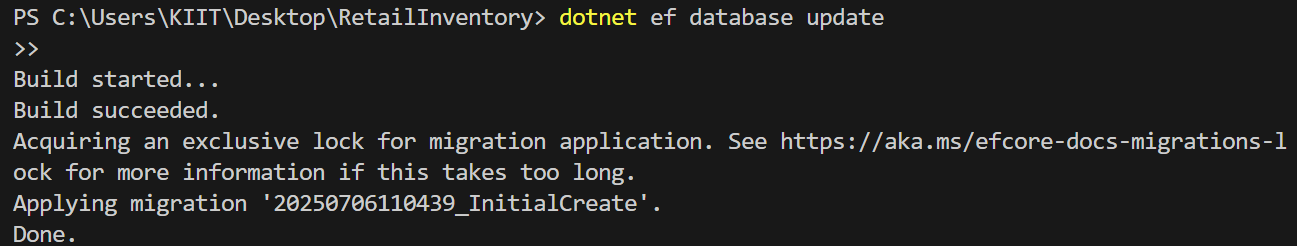
**Lab 3: Using EF Core CLI to Create and Apply Migrations.**

**#Install EF Core CLI**



**#Create the First Migration**





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

**#Program.cs**

using System;

using System.Threading.Tasks;

class Program

{

    static async Task Main()

    {

        using var context = new AppDbContext();

        var electronics = new Category { Name = "Electronics" };

        var groceries   = new Category { Name = "Groceries"  };

        await context.Categories.AddRangeAsync(electronics, groceries);

        var product1 = new Product { Name = "Laptop",   Price = 75000, Category = electronics };

        var product2 = new Product { Name = "Rice Bag", Price = 1200,  Category = groceries   };

        await context.Products.AddRangeAsync(product1, product2);

        await context.SaveChangesAsync();

        Console.WriteLine("Data inserted successfully!");

    }

}

**dotnet run**



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

**#Program.cs**

using System;

using System.Threading.Tasks;

using Microsoft.EntityFrameworkCore;

class Program

{

    static async Task Main()

    {

        using var context = new AppDbContext();

        var products = await context.Products.ToListAsync();

        Console.WriteLine("All Products:");

        foreach (var p in products)

            Console.WriteLine($"{p.Name} - ₹{p.Price}");

        var product = await context.Products.FindAsync(1);

        Console.WriteLine($"Found Product by ID 1: {product?.Name}");

        var expensive = await context.Products

                                     .FirstOrDefaultAsync(p => p.Price > 50000);

        Console.WriteLine($"Expensive Product: {expensive?.Name}");

    }

}

