Supersetid-6361766

**WEEK 3 HandsOn**

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

**Program.cs:-**

using Microsoft.EntityFrameworkCore;

using RetailInventory.Models;

using var db = new AppDbContext();

if (!db.Categories.Any())

{

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

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

db.Categories.AddRange(electronics, groceries);

db.Products.AddRange(

new Product { Name = "Laptop", Quantity = 10, Category = electronics },

new Product { Name = "Headphones", Quantity = 25, Category = electronics },

new Product { Name = "Apple", Quantity = 50, Category = groceries }

);

db.SaveChanges();

}

var products = db.Products.Include(p => p.Category).ToList();

Console.WriteLine($"{"ID",3} {"Name",-15} {"Qty",5} {"Category",-10}");

Console.WriteLine(new string('-', 40));

foreach (var p in products)

{

Console.WriteLine($"{p.ProductId,3} {p.Name,-15} {p.Quantity,5} {p.Category.Name,-10}");

}

Under .csproj file:-

< Project Sdk = "Microsoft.NET.Sdk" >

< PropertyGroup >

< OutputType > Exe </ OutputType >

< TargetFramework > net8.0 </ TargetFramework >

< ImplicitUsings > enable </ ImplicitUsings >

< Nullable > enable </ Nullable >

</ PropertyGroup >

< ItemGroup >

< PackageReference Include = "Microsoft.EntityFrameworkCore.Design" Version = "9.0.6" >

< IncludeAssets > runtime; build; native; contentfiles; analyzers; buildtransitive </ IncludeAssets >

< PrivateAssets > all </ PrivateAssets >

</ PackageReference >

< PackageReference Include = "Microsoft.EntityFrameworkCore.SqlServer" Version = "9.0.6" />

< PackageReference Include = "Microsoft.EntityFrameworkCore.Tools" Version = "9.0.6" >

< IncludeAssets > runtime; build; native; contentfiles; analyzers; buildtransitive </ IncludeAssets >

< PrivateAssets > all </ PrivateAssets >

</ PackageReference >

</ ItemGroup >

</ Project >

App DBcontext.cs:-

public class AppDbContext : DbContext

{

public DbSet<Product> Products => Set<Product>();

public DbSet<Category> Categories => Set<Category>();

protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)

{

optionsBuilder.UseSqlServer("Server=localhost\\SQLEXPRESS;Database=RetailInventoryDb;Trusted\_Connection=True;Encrypt=False;");

}

}

**Models:-**

category.cs

public class AppDbContext : DbContext

{

public DbSet<Product> Products => Set<Product>();

public DbSet<Category> Categories => Set<Category>();

protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)

{

optionsBuilder.UseSqlServer("Server=localhost\\SQLEXPRESS;Database=RetailInventoryDb;Trusted\_Connection=True;Encrypt=False;");

}

}

**Product.cs:-**

namespace RetailInventory.Models

{

public class Product

{

public int ProductId { get; set; }

public string Name { get; set; } = string.Empty;

public int Quantity { get; set; }

public int CategoryId { get; set; }

public Category Category { get; set; } = null!;

}

}

InitialCreate.Designer.cs

using Microsoft.EntityFrameworkCore;

using Microsoft.EntityFrameworkCore.Infrastructure;

using Microsoft.EntityFrameworkCore.Metadata;

using Microsoft.EntityFrameworkCore.Migrations;

using Microsoft.EntityFrameworkCore.Storage.ValueConvers;

namespace RetailInventory.Migrations

{

[DbContext(typeof(AppDbContext))]

[Migration("20250704075822\_InitialCreate")]

partial class InitialCreate

{

/// <inheritdoc />

protected override void BuildTargetModel(ModelBuilder modelBuilder)

{

#pragma warning disable 612, 618

modelBuilder

.HasAnnotation("ProductVersion", "9.0.6")

.HasAnnotation("Relational:MaxIdentifierLength", 128);

SqlServerModelBuilderExtensions.UseIdentityColumns(modelBuilder);

modelBuilder.Entity("RetailInventory.Models.Category", b =>

{

b.Property<int>("CategoryId")

.ValueGeneratedOnAdd()

.HasColumnType("int");

SqlServerPropertyBuilderExtensions.UseIdentityColumn(b.Property<int>("CategoryId"));

b.Property<string>("Name")

.IsRequired()

.HasColumnType("nvarchar(max)");

b.HasKey("CategoryId");

b.ToTable("Categories");

});

modelBuilder.Entity("RetailInventory.Models.Product", b =>

{

b.Property<int>("ProductId")

.ValueGeneratedOnAdd()

.HasColumnType("int");

SqlServerPropertyBuilderExtensions.UseIdentityColumn(b.Property<int>("ProductId"));

b.Property<int>("CategoryId")

.HasColumnType("int");

b.Property<string>("Name")

.IsRequired()

.HasColumnType("nvarchar(max)");

b.Property<int>("Quantity")

.HasColumnType("int");

b.HasKey("ProductId");

b.HasIndex("CategoryId");

b.ToTable("Products");

});

modelBuilder.Entity("RetailInventory.Models.Product", b =>

{

b.HasOne("RetailInventory.Models.Category", "Category")

.WithMany("Products")

.HasForeignKey("CategoryId")

.OnDelete(DeleteBehavior.Cascade)

.IsRequired();

b.Navigation("Category");

});

modelBuilder.Entity("RetailInventory.Models.Category", b =>

{

b.Navigation("Products");

});

#pragma warning restore 612, 618

}

}

}

**InitialCreate.cs**

using Microsoft.EntityFrameworkCore.Migrations;

#nullable disable

namespace RetailInventory.Migrations

{

/// <inheritdoc />

public partial class InitialCreate : Migration

{

/// <inheritdoc />

protected override void Up(MigrationBuilder migrationBuilder)

{

migrationBuilder.CreateTable(

name: "Categories",

columns: table => new

{

CategoryId = table.Column<int>(type: "int", nullable: false)

.Annotation("SqlServer:Identity", "1, 1"),

Name = table.Column<string>(type: "nvarchar(max)", nullable: false)

},

constraints: table =>

{

table.PrimaryKey("PK\_Categories", x => x.CategoryId);

});

migrationBuilder.CreateTable(

name: "Products",

columns: table => new

{

ProductId = table.Column<int>(type: "int", nullable: false)

.Annotation("SqlServer:Identity", "1, 1"),

Name = table.Column<string>(type: "nvarchar(max)", nullable: false),

Quantity = table.Column<int>(type: "int", nullable: false),

CategoryId = table.Column<int>(type: "int", nullable: false)

},

constraints: table =>

{

table.PrimaryKey("PK\_Products", x => x.ProductId);

table.ForeignKey(

name: "FK\_Products\_Categories\_CategoryId",

column: x => x.CategoryId,

principalTable: "Categories",

principalColumn: "CategoryId",

onDelete: ReferentialAction.Cascade);

});

migrationBuilder.CreateIndex(

name: "IX\_Products\_CategoryId",

table: "Products",

column: "CategoryId");

}

/// <inheritdoc />

protected override void Down(MigrationBuilder migrationBuilder)

{

migrationBuilder.DropTable(

name: "Products");

migrationBuilder.DropTable(

name: "Categories");

}

}

}

Output:-

ID Name Qty Category

------------------------------------

1 Laptop 10 Electronics

2 Headphones 25 Electronics

3 Apple 50 Groceries