In ClassLibrary Project

**Category.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ClassLibrary1

{

public class Category

{

public int CategoryId { get; set; }

public string CategoryName { get; set; }

public List<Product> Products { get; set; } //one to many---naviagtion property

}

}

**Product.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ClassLibrary1

{

public class Product

{

public int ProductId { get; set; }

public string ProductName { get; set; }

public int Price { get; set; }

public int CategoryId { get; set; } //foreign key

public Category Category { get; set; } //one to one---naviagtion property

}

}

**ShoppingContext.cs**

using Microsoft.EntityFrameworkCore;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ClassLibrary1

{

public class ShoppingContext:DbContext

{

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

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

protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)

{

optionsBuilder.UseSqlServer(@"Server=(localdb)\MSSQLLocalDB;Database=sejal\_pre;Trusted\_Connection=True");

}

}

}

**Inside ConsoleApp Project**

**Program.cs**

using ClassLibrary1;

namespace ConsoleApp1

{

internal class Program

{

static ShoppingContext context;

static Program()

{

context = new ShoppingContext();

}

static void Main(string[] args)

{

//AddCategory();

// UpdateCategory();

// DeleteCategory();

// getAllCategories();

// GetCategoryByID();

//AddProduct();

// UpdateProduct();

//DeleteProduct();

// getAllProducts();

//getAllProductsByCategory();

getProductById();

Console.ReadKey();

}

public static void AddCategory()

{

Console.WriteLine("Enter Category Name:");

string cnm=Console.ReadLine();

Category c = new Category();

c.CategoryName = cnm;

context.Categories.Add(c);

context.SaveChanges();

Console.WriteLine("Category inserted successfully");

}

public static void UpdateCategory()

{

Console.WriteLine("Enter Category Id:");

int cid=int.Parse(Console.ReadLine());

Console.WriteLine("Enter Category Name:");

string cnm = Console.ReadLine();

Category c = context.Categories.Find(cid);

if (c == null)

{

Console.WriteLine("Category not found");

}

else

{

c.CategoryName = cnm;

context.SaveChanges();

Console.WriteLine("Category Updated");

}

}

public static void DeleteCategory()

{

Console.WriteLine("Enter Category Id:");

int cid = int.Parse(Console.ReadLine());

Category c = context.Categories.Find(cid);

if (c == null)

{

Console.WriteLine("Category not found");

}

else

{

context.Categories.Remove(c);

context.SaveChanges();

Console.WriteLine("Category Deleted");

}

}

public static void getAllCategories()

{

List<Category> clist=context.Categories.ToList();

foreach (Category c in clist)

{

Console.WriteLine(c.CategoryId + "\t" + c.CategoryName);

}

}

public static void GetCategoryByID()

{

Console.WriteLine("Enter Category Id:");

int cid= int.Parse(Console.ReadLine());

Category c=context.Categories.Find(cid);

if (c == null)

{

Console.WriteLine("Category Not Found");

}

else

{

Console.WriteLine("Category Name:" + c.CategoryName);

}

}

public static void AddProduct()

{

Console.WriteLine("Enter Product Name:");

string pnm = Console.ReadLine();

Console.WriteLine("Enter Product Price:");

int pr = int.Parse(Console.ReadLine());

Console.WriteLine("Enter Categoty Id:");

int cid = int.Parse(Console.ReadLine());

Product p = new Product();

p.ProductName = pnm;

p.Price = pr;

p.CategoryId = cid;

context.Products.Add(p);

context.SaveChanges();

Console.WriteLine("Product Inserted");

}

public static void UpdateProduct()

{

Console.WriteLine("Enter Product Id:");

int pid = int.Parse(Console.ReadLine());

Console.WriteLine("Enter Product Name:");

string pnm = Console.ReadLine();

Console.WriteLine("Enter Product Price:");

int pr = int.Parse(Console.ReadLine());

Console.WriteLine("Enter Categoty Id:");

int cid = int.Parse(Console.ReadLine());

Product p = context.Products.Find(pid);

if (p != null)

{

p.ProductName = pnm;

p.Price = pr;

p.CategoryId = cid;

context.SaveChanges();

Console.WriteLine("Product Updated");

}

else

{

Console.WriteLine("Product Not Found");

}

}

public static void DeleteProduct()

{

Console.WriteLine("Enter Product Id:");

int pid = int.Parse(Console.ReadLine());

Product p = context.Products.Find(pid);

if (p != null)

{

context.Products.Remove(p);

context.SaveChanges();

Console.WriteLine("Product Deleted");

}

else

{

Console.WriteLine("Product Not Found");

}

}

public static void getAllProducts()

{

List<Product> plist= context.Products.ToList();

foreach(Product p in plist)

{

Category c = context.Categories.Find(p.CategoryId);

Console.WriteLine(p.ProductId + "\t" + p.ProductName + "\t" + p.Price + "\t" + p.CategoryId + "\t" + c.CategoryName) ;

}

}

public static void getAllProductsByCategory()

{

Console.WriteLine("Enter Category Id:");

int cid= int.Parse(Console.ReadLine());

List<Product> plist = context.Products.Where(p => p.CategoryId == cid).ToList();

foreach (Product p in plist)

{

Category c = context.Categories.Find(p.CategoryId);

Console.WriteLine(p.ProductId + "\t" + p.ProductName + "\t" + p.Price + "\t" + p.CategoryId + "\t" + c.CategoryName);

}

}

public static void getProductById()

{

Console.WriteLine("Enter Product Id:");

int pid = int.Parse(Console.ReadLine());

Product p=context.Products.Find(pid);

if (p != null)

{

Category c = context.Categories.Find(pid);

Console.WriteLine(p.ProductId + "\t" + p.ProductName + "\t" + p.Price + "\t" + p.CategoryId + "\t" + c.CategoryName);

}

else

{

Console.WriteLine("Product Not Found");

}

}

}

}