using Microsoft.EntityFrameworkCore;

namespace PizzaStore.Models

{

    class PizzaDb : DbContext

    {

        public PizzaDb(DbContextOptions<PizzaDb> options) : base(options) {

        }

// IMPORTANTISIMO DEBE LLAMARSE IGUAL A LAS TABLAS

        public DbSet<Pizza> Pizzas { get; set; }

        public DbSet<Persona> Persona { get; set; }

        public DbSet<Books> Books { get; set; }

    }

    public class Persona

    {

      public int id { get; set; } = 1;

      public string? Nombre { get; set; }

    }

    public class Books

    {

      public int Id { get; set; } = 1;

      public string? Titulo { get; set; }

      public int? Author { get; set; }

  }

}

using Microsoft.OpenApi.Models;

using Microsoft.EntityFrameworkCore;

using PizzaStore.Models;

var builder = WebApplication.CreateBuilder(args);

builder.Services.AddEndpointsApiExplorer();

ConfigurationManager configuration = builder.Configuration;

var conn = configuration.GetConnectionString("Conexion");

builder.Services.AddDbContext<PizzaDb>(options => options.UseSqlServer(conn));

builder.Services.AddSwaggerGen(c =>

    {

          c.SwaggerDoc("v1", new OpenApiInfo {

              Title = "PizzaStore API", Description = "Making Pizzas", Version = "v1"

          });

     }

);

var app = builder.Build();

app.UseSwagger(); app.UseSwaggerUI(c =>

    {

          c.SwaggerEndpoint("/swagger/v1/swagger.json", "PizzaStore API V1");

     }

);

app.MapGet("/LINQ", (PizzaDb db) =>

{

      List<Persona> Persona = db.Persona.ToList<Persona>();

      List<Books> Libros = db.Books1.ToList<Books>();

      var query = from A in Libros

                  join B in Persona

                   on A.Author equals B.id

            select new

            {

                B.Nombre,

                A.Titulo

           };

        return query;

});

      var Persona = db.Authors1.ToListAsync();