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

**Files**

1. **Category.cs**

using System.Collections.Generic;

namespace RetailInventory.Models

{

    public class Category

    {

        public int Id { get; set; }

        public string Name { get; set; }

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

    }

}

1. **Product.cs**

namespace RetailInventory.Models

{

    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; }

    }

}

1. **AppDbContext.cs**

using Microsoft.EntityFrameworkCore;

using RetailInventory.Models;

namespace RetailInventory

{

    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=localhost;Database=RetailDb;Trusted\_Connection=True;TrustServerCertificate=True;");

        }

    }

}

1. **appsettings.json**

{

  "ConnectionStrings": {

    "DefaultConnection": "Server=localhost;Database=RetailDb;Trusted\_Connection=True;TrustServerCertificate=True;"

  }

}

1. **Program.cs**

using Microsoft.Extensions.Configuration;

var config = new ConfigurationBuilder()

    .AddJsonFile("appsettings.json")

    .Build();

var connectionString = config.GetConnectionString("DefaultConnection");

Console.WriteLine($"Connection string: {connectionString}");