To create a Class Library with Entity Framework Core, use the edit project, option, and then insert the yellow highlighted text to allow the use of Entity Framework Core:

<TargetFramework>netcoreapp1.0</TargetFramework>

<AssemblyName>GR.Data</AssemblyName>

<PackageId>GR.Data</PackageId>

<PackageTargetFallback>$(PackageTargetFallback);dotnet5.6;portable-net45+win8</PackageTargetFallback>

<GenerateAssemblyConfigurationAttribute>false</GenerateAssemblyConfigurationAttribute>

<GenerateAssemblyCompanyAttribute>false</GenerateAssemblyCompanyAttribute>

<GenerateAssemblyProductAttribute>false</GenerateAssemblyProductAttribute>

</PropertyGroup>

<ItemGroup>

<PackageReference Include="Microsoft.EntityFrameworkCore.SqlServer" Version="1.0.3" />

<PackageReference Include="Microsoft.EntityFrameworkCore.Tools" Version="1.0.0" />

</ItemGroup>

<ItemGroup>

<DotNetCliToolReference Include="Microsoft.EntityFrameworkCore.Tools.DotNet" Version="1.0.0" />

</ItemGroup>

</Project>

public class HomeController : Controller

{

private IRepository<Author> repoAuthor;

private IRepository<Book> repoBook;

public HomeController(IRepository<Author> repoAuthor, IRepository<Book> repoBook)

{

this.repoAuthor = repoAuthor;

this.repoBook = repoBook;

}

[HttpGet]

public IActionResult Index()

{

List<AuthorListingViewModel> model = new List<AuthorListingViewModel>();

repoAuthor.GetAll().ToList().ForEach(a => {

AuthorListingViewModel author = new AuthorListingViewModel

{

Id = a.Id,

Name = a.LastName + "," + a.FirstName,

Email = a.Email

};

author.TotalBooks = repoBook.GetAll().Count(x => x.AuthorId == a.Id);

model.Add(author);

});

return View("Index", model);

}

public class ApplicationContext : DbContext

{

public ApplicationContext(DbContextOptions<ApplicationContext> options) : base(options) { }

protected override void OnModelCreating(ModelBuilder modelBuilder)

{

base.OnModelCreating(modelBuilder);

}

public DbSet<Author> Authors { get; set; }

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

}

{

"ConnectionStrings": {

"DefaultConnection": "Server=(localdb)\\mssqllocaldb;Database=GRepoDb;Trusted\_Connection=True;MultipleActiveResultSets=true"

},

"ApplicationInsights": {

"InstrumentationKey": ""

},

"Logging": {

"IncludeScopes": false,

"LogLevel": {

"Default": "Debug",

"System": "Information",

"Microsoft": "Information"

}

}

}

@model IEnumerable<AuthorListingViewModel>

@foreach(var item in Model)

{

<tr>

<td>@item.Id</td>

<td>@item.Name</td>

<td>@item.TotalBooks.ToString()</td>

</tr>

}

@using GenericRepository

@using GenericRepository.Models

@using GenericRepository.Data

@using GenericRepository.Entities

@using GenericRepository.Services

@addTagHelper \*, Microsoft.AspNetCore.Mvc.TagHelpers

services.AddDbContext<ApplicationContext>(options => options.UseSqlServer(Configuration.GetConnectionString("DefaultConnection")));

services.AddScoped(typeof(IRepository<>), typeof(Repository<>));