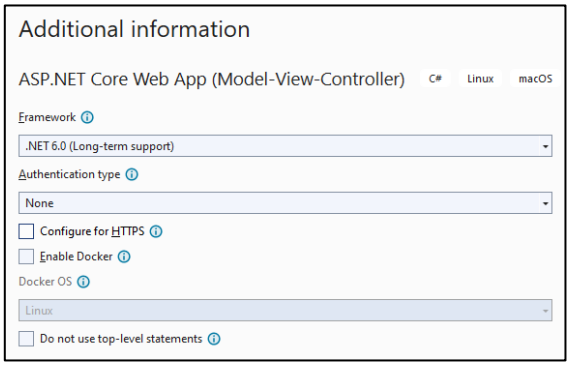
1. Criar um Projeto

1. Adicionar 1 projeto com o template: ASP.NET Core Web Application (Model-View-Controller)

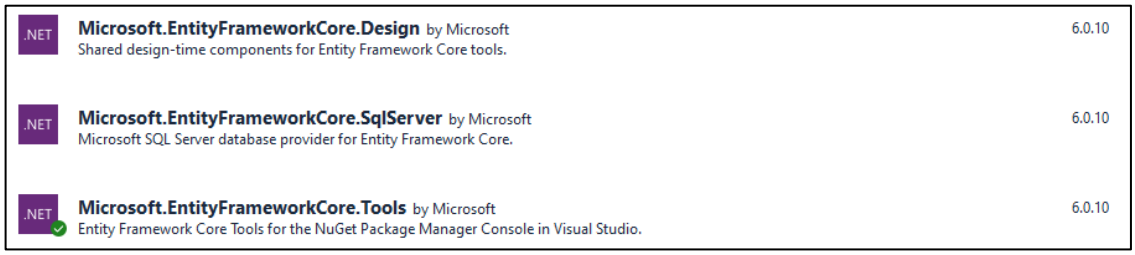


2. Instalar EF Core

1. Abrir Package Manager Console (PM)

```
Install-Package Microsoft.EntityFrameworkCore.SqlServer -Version 6.0.10
Install-Package Microsoft.EntityFrameworkCore.Tools -Version 6.0.10
Install-Package Microsoft.EntityFrameworkCore.Design -Version 6.0.10
```

2. Verificar os packages:



3. Criar os Models

1. Criar a classe Contact.cs na pasta Models:

Contact.cs

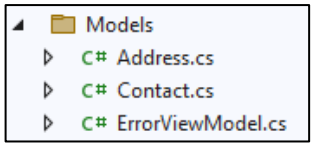
```
public partial class Contact
{
    public Contact()
    {
        Address = new HashSet<Address>();
    }

    public int ContactId { get; set; }

    [Required(ErrorMessage = "Required.")]
    [MaxLength(100)]
    public string ContactName { get; set; } = null!;

    [Required(ErrorMessage = "Required.")]
    [MaxLength(255)]
    public string Email { get; set; } = null!;
```

```
        [MaxLength(12)]
        public string? Phone { get; set; }

        public virtual ICollection<Address> Address { get; set; }

    }

}
```

2. Criar a classe Address.cs na pasta Models:

```
Address.cs
using System.ComponentModel.DataAnnotations;

namespace D07_EFCore_CF.Models
{

    public partial class Address
    {

        public int AddressId { get; set; }
        public int ContactId { get; set; }

        [Required(ErrorMessage = "Required.")]
        [MaxLength(150)]
        public string ContactAddress { get; set; } = null!;

        [Required(ErrorMessage = "Required.")]
        [MaxLength(8)]
        public string ZipCode { get; set; } = null!;

        [Required(ErrorMessage = "Required.")]
        [MaxLength(70)]
        public string City { get; set; } = null!;

        public virtual Contact Contact { get; set; } = null!;

    }

}
```

3. Verificar os modelos:



4. Configurar a ConnectionString

1. Configurar a ConnectionString no ficheiro appsettings.json:

```
appsettings.json
{
  "Logging": {
    "LogLevel": {
      "Default": "Information",
      "Microsoft.AspNetCore": "Warning"
    }
  },
  "AllowedHosts": "*"
  "ConnectionStrings": {
    "ContactDB_EFCore_CodeFirst_ConnectionString":
"Server=mrspcxl;Database=ContactDB_EFCore_CodeFirst;Trusted_Connection=True"
  }
}
```

5. Criar o DBContext

1. Criar a pasta DAL na raíz do projeto.

2. Adicionar na pasta DAL a classe ContactDB_EFCore_CodeFirst_DBContext.cs:

ContactDB_EFCore_CodeFirst_DBContext.cs

```
using D07_EFCore_CF.Models;
using Microsoft.EntityFrameworkCore;

namespace D07_EFCore_CF.DAL
{

    public partial class ContactDB_EFCore_CodeFirst_DBContext : DbContext
    {

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

        public virtual DbSet<Address> Address { get; set; } = null!;
        public virtual DbSet<Contact> Contact { get; set; } = null!;

    }

}
```

6. Registar a EF como Serviço

1. Registar o serviço da EF no program.cs:

Program.cs

```
using D07_EFCore_CF.DAL;
using Microsoft.EntityFrameworkCore;

var builder = WebApplication.CreateBuilder(args);

// TODO MRS: ler o nome da connection string do appsettings.json
var connectionString =
builder.Configuration.GetConnectionString("ContactDB_EFCore_CodeFirst_ConnectionString");

// TODO MRS: registar o serviço da EF
builder.Services.AddDbContext<ContactDB_EFCore_CodeFirst_DBContext>(options =>
options.UseSqlServer(connectionString));

// Add services to the container.
builder.Services.AddControllersWithViews();

var app = builder.Build();
```

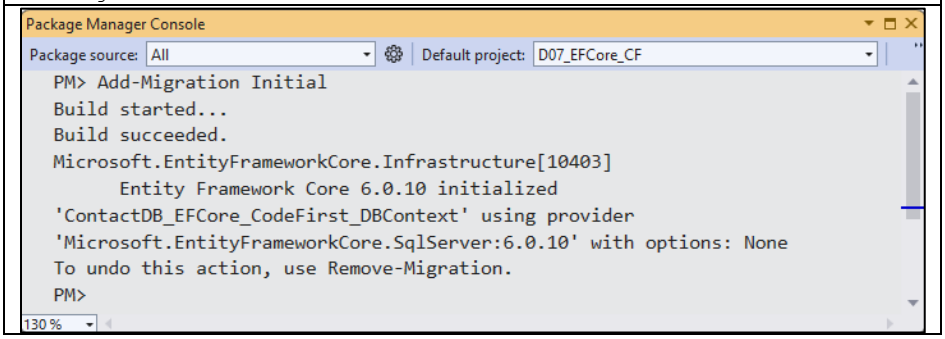
7. Criar a BD com Migrations Manuais

1. Lançar o Package Manager Console.

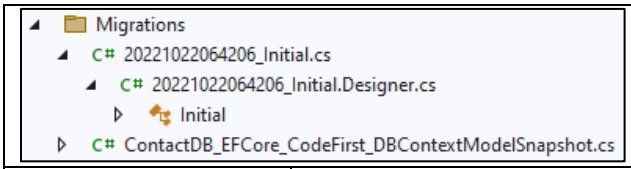
2. Selecionar o projeto atual no PM.

3. Criar a migration inicial:

```
Add-Migration Initial
```



4. É criada a pasta Migrations:

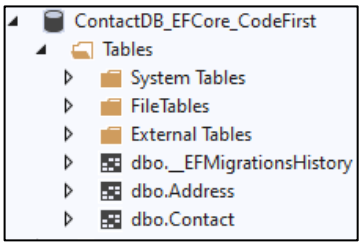


_.cs	Migration criada e identificada com um timestamp. Inclui os métodos Up() e Down(), para criar e remover os objetos da BD.
_.Designer.cs	Ficheiro com a metadata da migration.
ModelSnapshot.cs	Snapshot do modelo atual, usado para verificar se houve alterações quando se criar a próxima migration.

3. Criar a bd:

```
Update-Database
```

4. Verificar no SQL Server Object Explorer a nova bd:



8. Alterações aos Modelos (se necessário)

1. Alterar a classe do modelo, acrescentando uma nova propriedade:

```
Address.cs
[Required(ErrorMessage = "Required.")]
[MaxLength(50)]
public string Country { get; set; } = null!;
```

2. Criar outra migration e atualizar a BD:

```
Add-Migration AddressAddColumn
Update-Database
```

3. Remover a última propriedade adicionada ao modelo.

4. Criar outra migration e atualizar a bd:

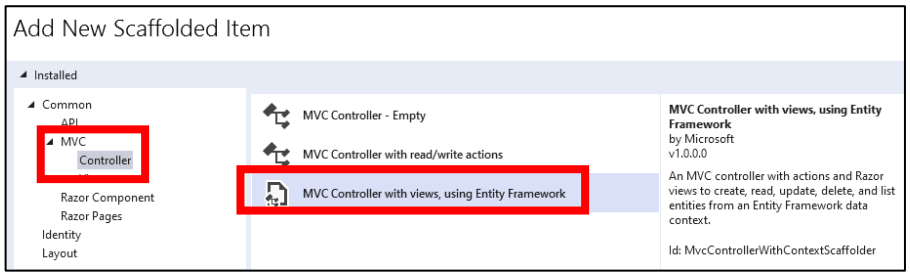
```
Add-Migration AddressDropColumn
Update-Database
```

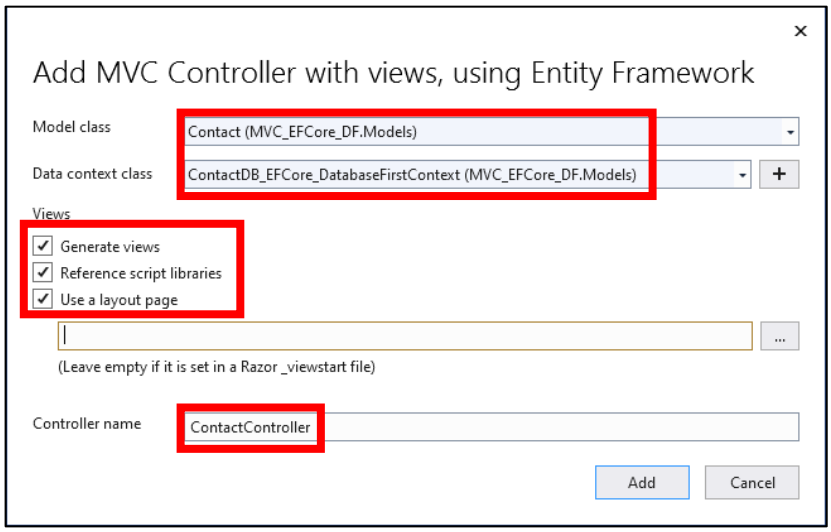
5. Aplicar uma migration específica

```
Update-Database -Migration AddressAddColumn
Update-Database
```

9. Criar os Controllers com Scaffolding

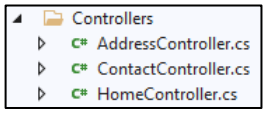
1. Criar o controlador para cada model: selecionar a pasta Controllers > Right click > Add > Controller > MVC Controller with views, using Entity Framework > Add:





2. Repetir a operação para todos os modelos.

3. Verificar os controladores:



10. Alterar o Layout

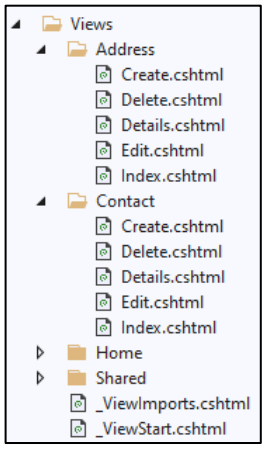
1. Alterar o layout e incluir os links, na navigation bar, para os novos controladores.

```
_Layout.cshtml
<li class="nav-item">
   <a class="nav-link text-dark" asp-area="" asp-controller="Contact" asp-action="Index">Contact</a>
</li>
<li class="nav-item">
   <a class="nav-link text-dark" asp-area="" asp-controller="Address" asp-action="Index">Address</a>
</li>
```

```
<!--TODO MRS: acrescentar links à nav bar-->
<li class="nav-item">
   <a class="nav-link text-dark" asp-area="" asp-controller="Contact" asp-action="Index">Contacts</a>
</li>
<li class="nav-item">
   <a class="nav-link text-dark" asp-area="" asp-controller="Address" asp-action="Index">Addresses</a>
</li>
```

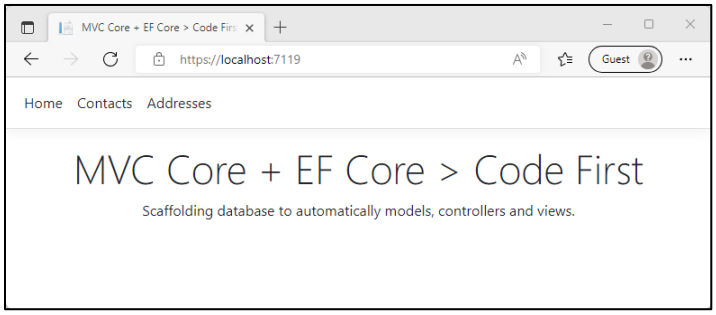
11. Verificar as Views

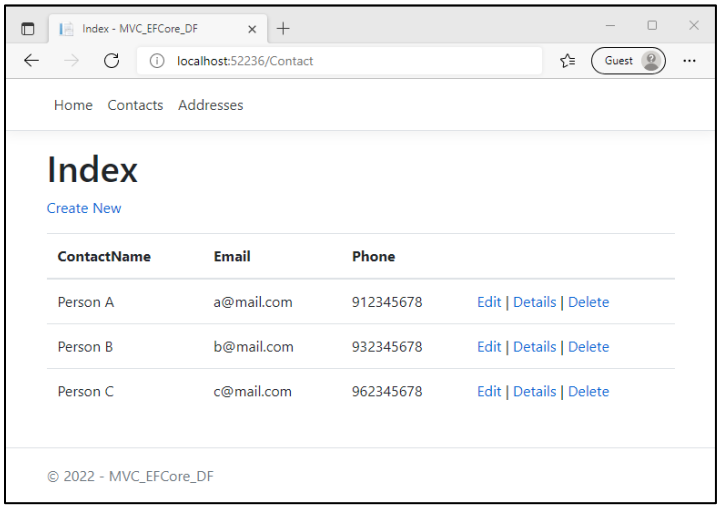
1. As vistas foram geradas por scaffolding:

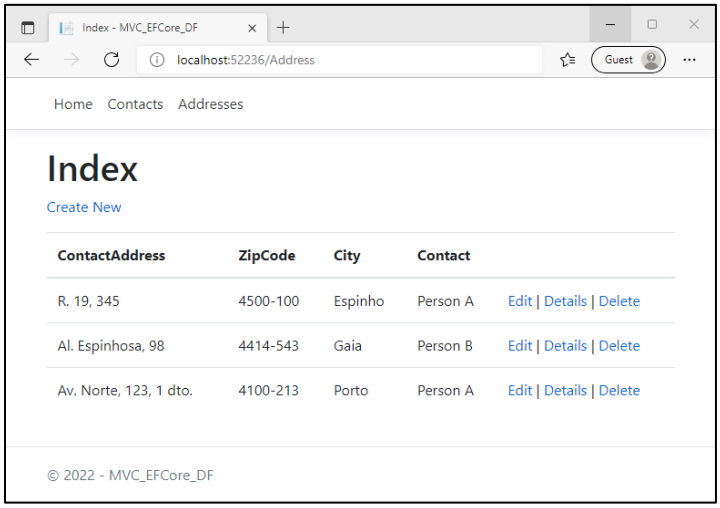


2. Fazer os ajustes necessários.

12. Testar a Aplicação




Index - MVC_EFCore_DF
localhost:52236/Contact
Guest
Home Contacts Addresses
Index
Create New
ContactName Email Phone
Person A a@mail.com 912345678 Edit | Details | Delete
Person B b@mail.com 932345678 Edit | Details | Delete
Person C c@mail.com 962345678 Edit | Details | Delete
© 2022 - MVC_EFCore_DF


Index - MVC_EFCore_DF
localhost:52236/Address
Guest
Home Contacts Addresses
Index
Create New
ContactAddress ZipCode City Contact
R. 19, 345 4500-100 Espinho Person A Edit | Details | Delete
Al. Espinhosa, 98 4414-543 Gaia Person B Edit | Details | Delete
Av. Norte, 123, 1 dto. 4100-213 Porto Person A Edit | Details | Delete
© 2022 - MVC_EFCore_DF