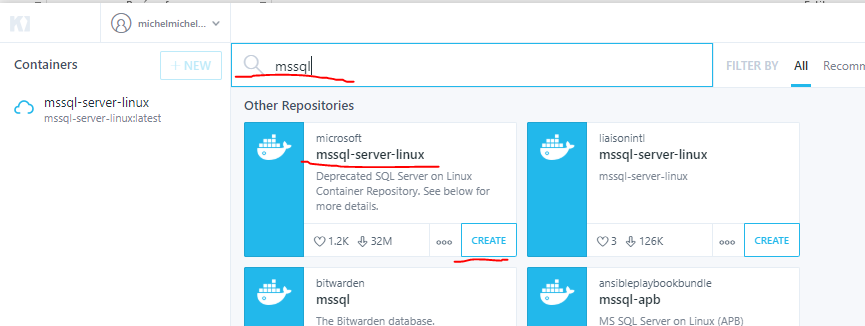
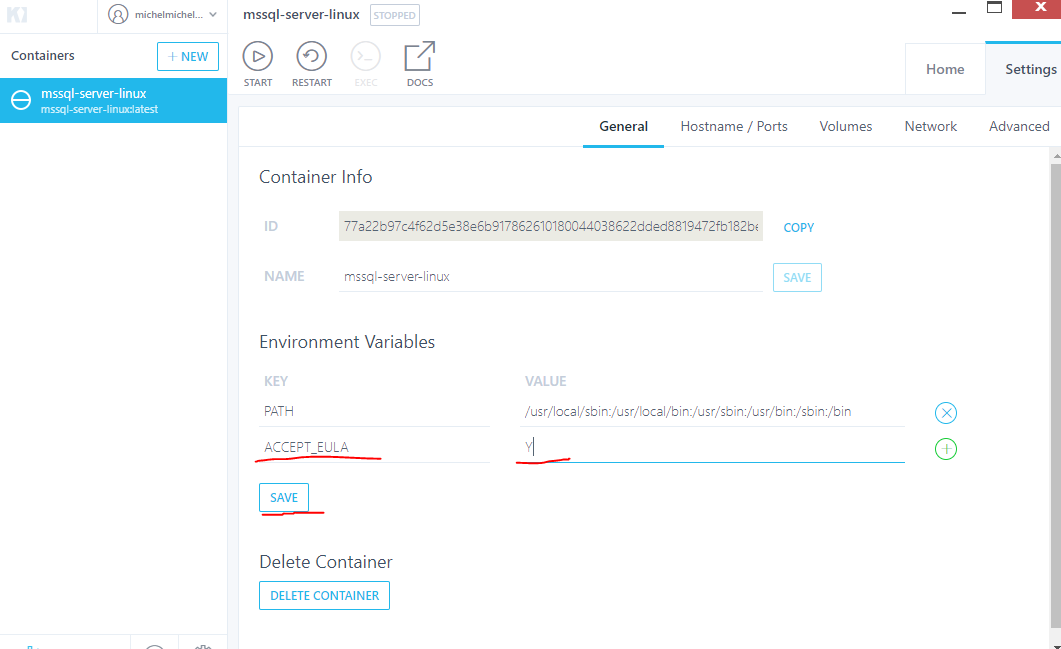
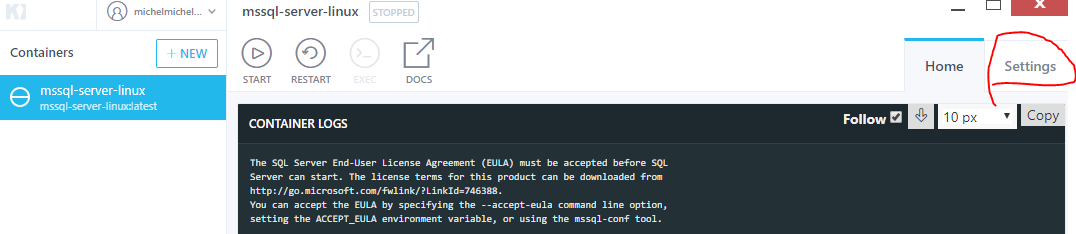
**Configurar docker para SQLServer**

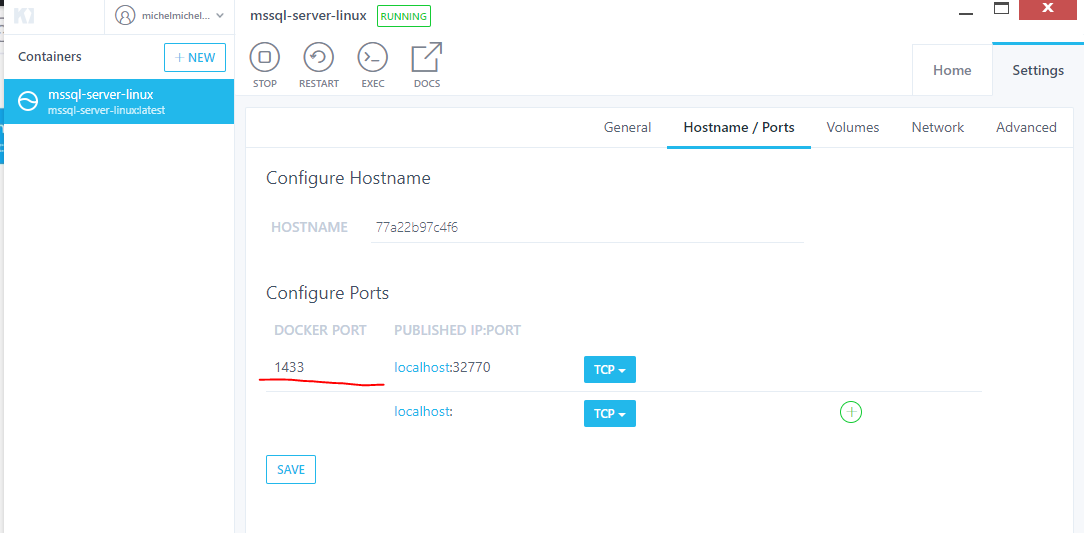
Baixe o docker e o Kitematic

Coloque o Kitematic na pasta > arquivo de programas > docker

Execute o Kitematic







"ConnectionStrings":{

    "connectionStrings":

"server=localhost, 1433; database=todos; User ID=SA; Password=1q2w3e4r!@#$"

  }

Iniciando projeto – Nome do projeto de Exemplo “Todo”

Criar pastas:

Todo.Domain

Todo.Domain.Api

Todo.Domain.Infra

Todo.Domain.Tests

Criar classlib/webapi/mstest

Cd Todo.Domain > dotnet new classlib

cd .\Todo.Domain.Api\ > dotnet new webapi

cd .\Todo.Domain.Infra\ > dotnet new classlib

cd .\Todo.Domain.Tests\ > dotnet new mstest

Criar solução

src > dotnet new sln -n Todo

Adicionar na solução

src> dotnet sln add .\Todo.Domain\

src> dotnet sln add .\Todo.Domain.Api\

src> dotnet sln add .\Todo.Domain.Tests\

src> dotnet sln add .\Todo.Domain.Infra\

Referencias:

O domain nunca referencia outros projetos, ele é sempre refereniado

src> cd .\Todo.Domain.Api\

src\Todo.Domain.Api> dotnet add reference ..\Todo.Domain\

src\Todo.Domain.Api> dotnet add reference ..\Todo.Domain.Infra\

src> cd .\Todo.Domain.Infra\

src\Todo.Domain.Infra> dotnet add reference ..\Todo.Domain\

cd .\Todo.Domain.Tests\

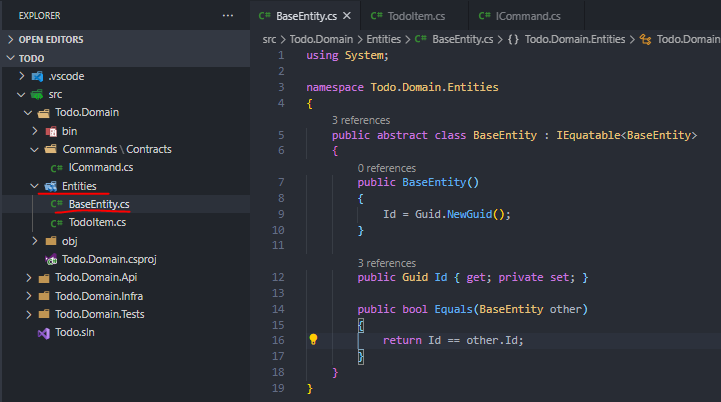
src\Todo.Domain.Tests> dotnet add reference ..\Todo.Domain\

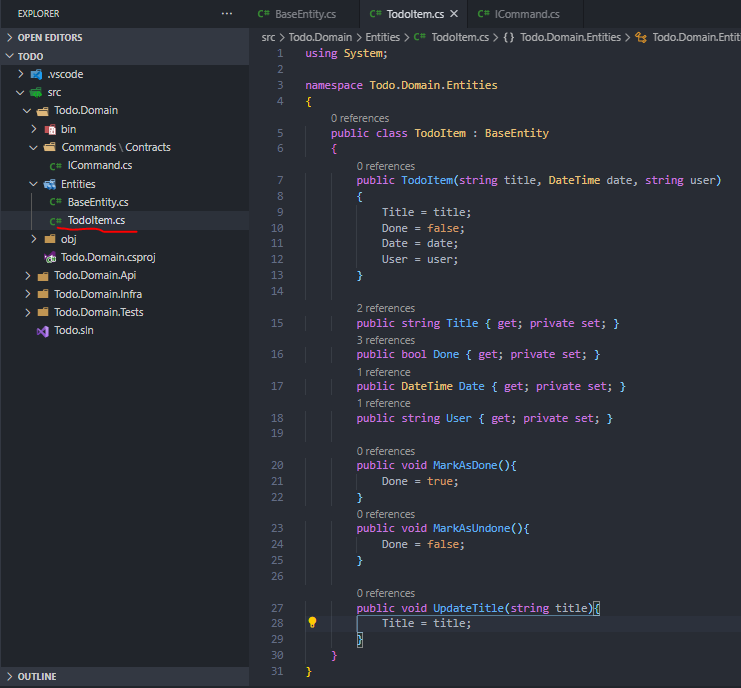
**Instalação – JWT**

src\Todo.Domain.Api> dotnet add package Microsoft.AspNetCore.Authentication.JwtBearer

**Confiar certificados para aplicações localhost**

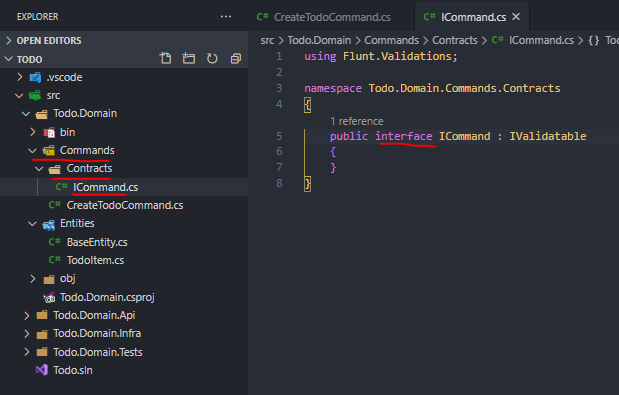
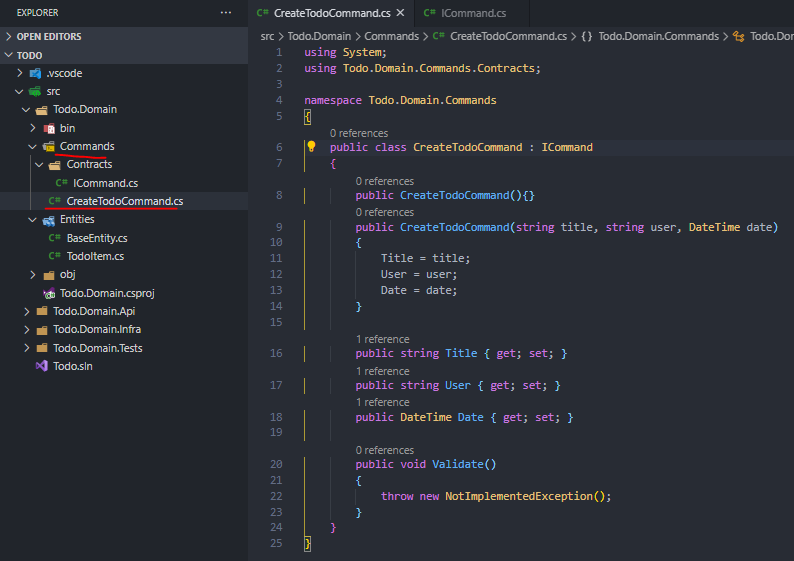
src > dotnet dev-certs https --trust

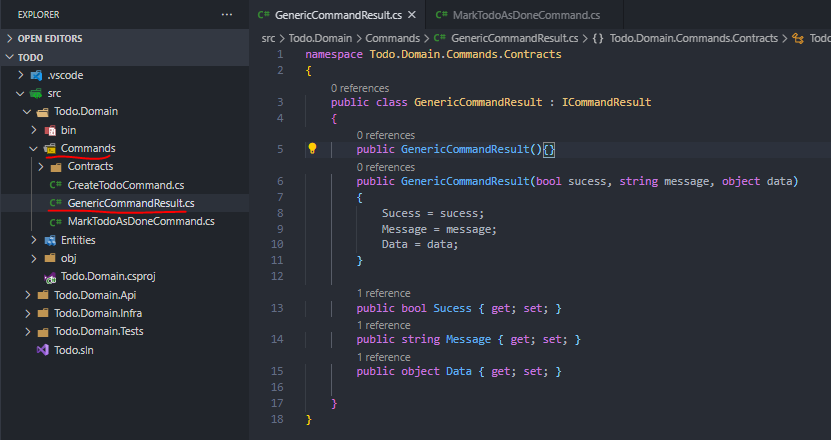
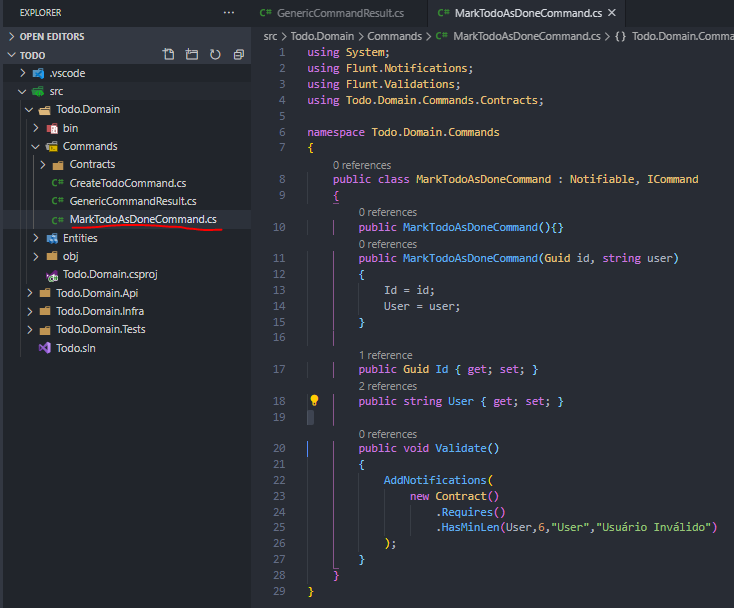
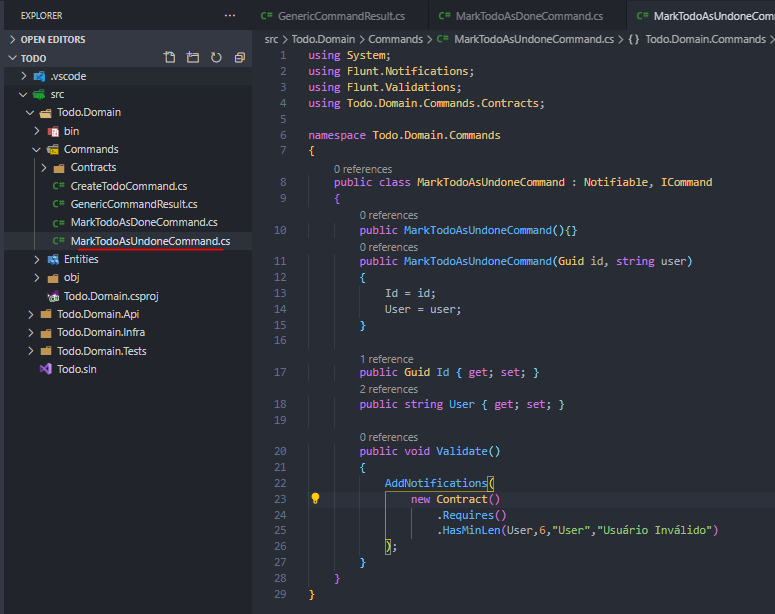
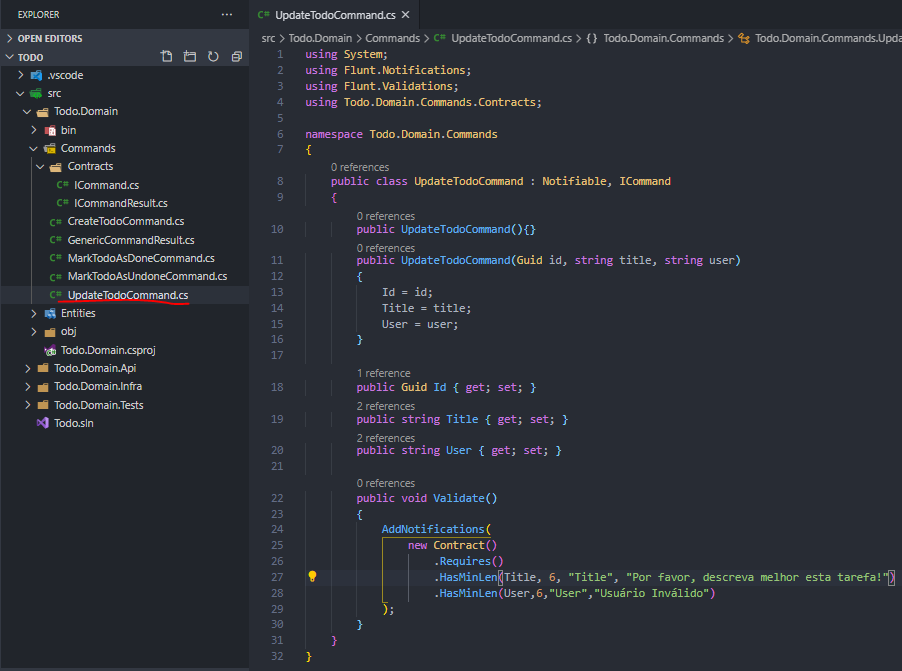
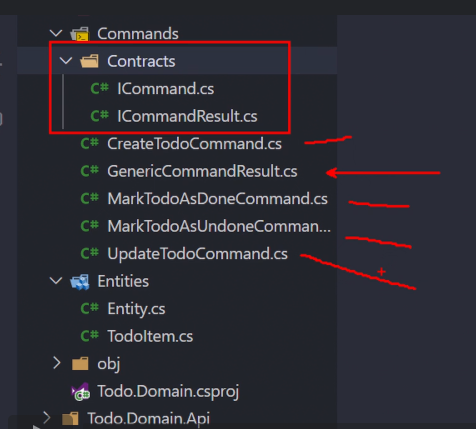




Adicionar o package FLUNT para fazer validações

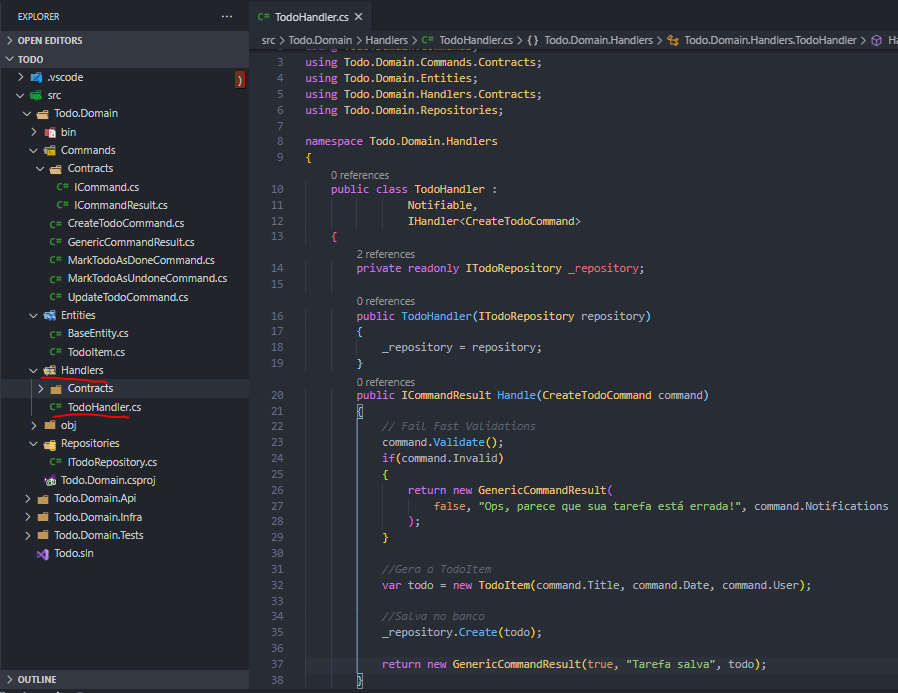
src\Todo.Domain> dotnet add package flunt

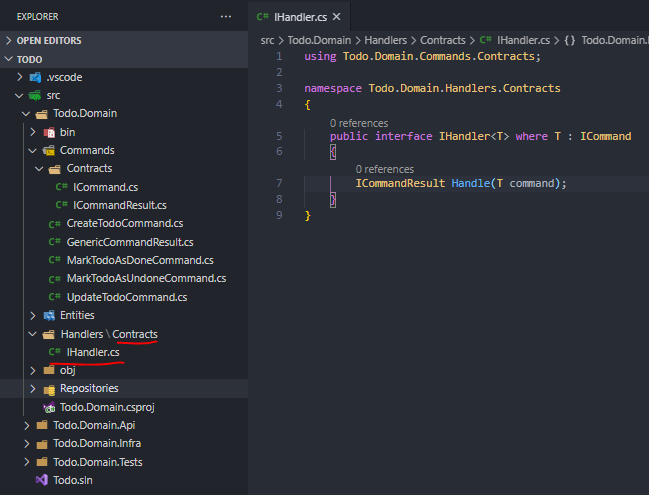
 

Se trabalhar com varios dominios, é recomendado criar um pasta Shared para conter esses arquivos, pois são comuns para todos os commands

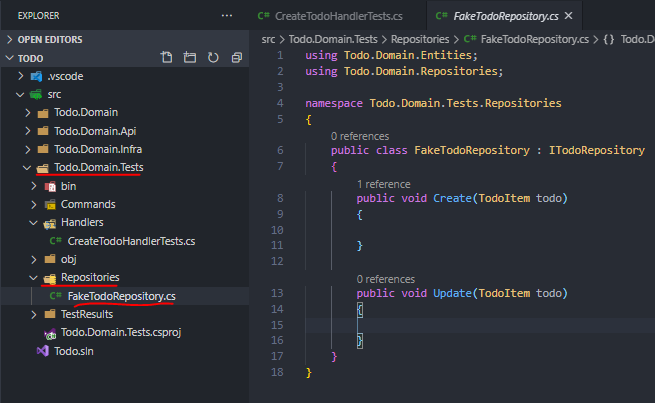
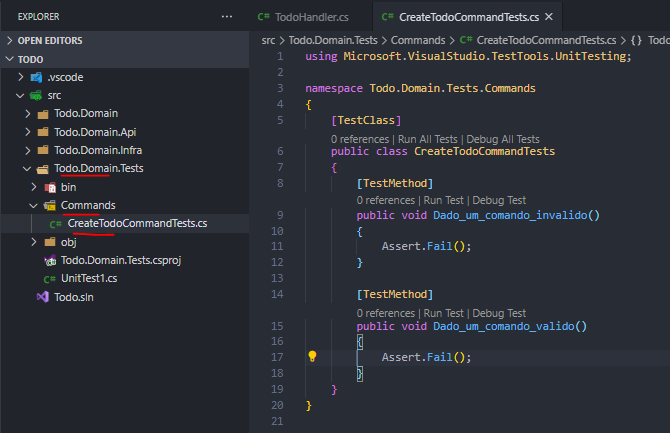
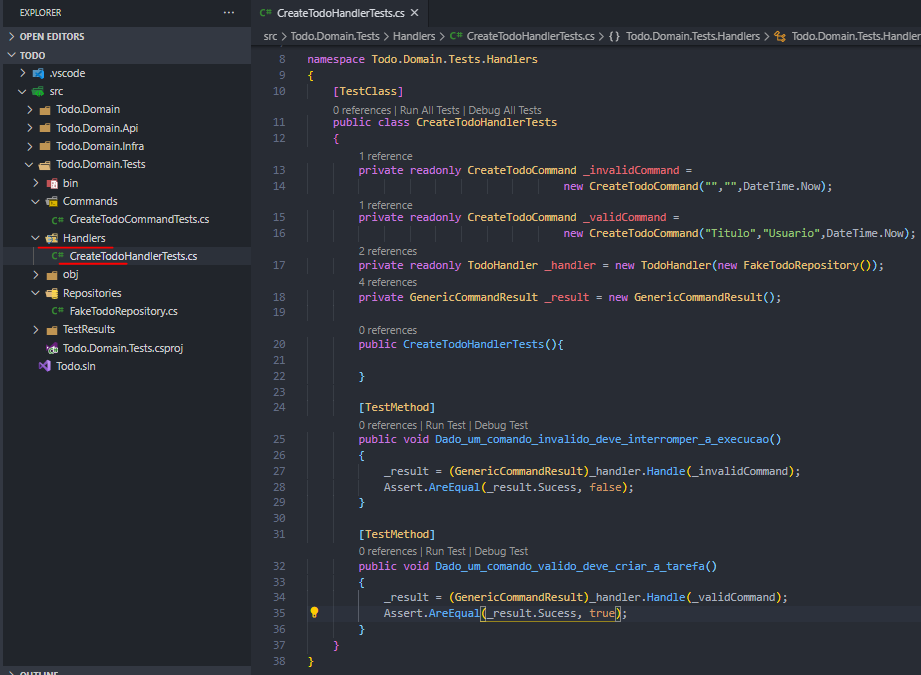
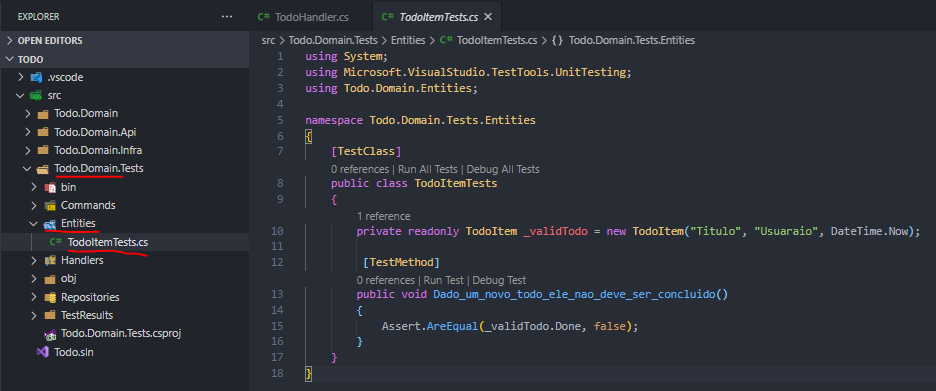
**Todo commando de entrada da aplicação fica contido nos Commands**



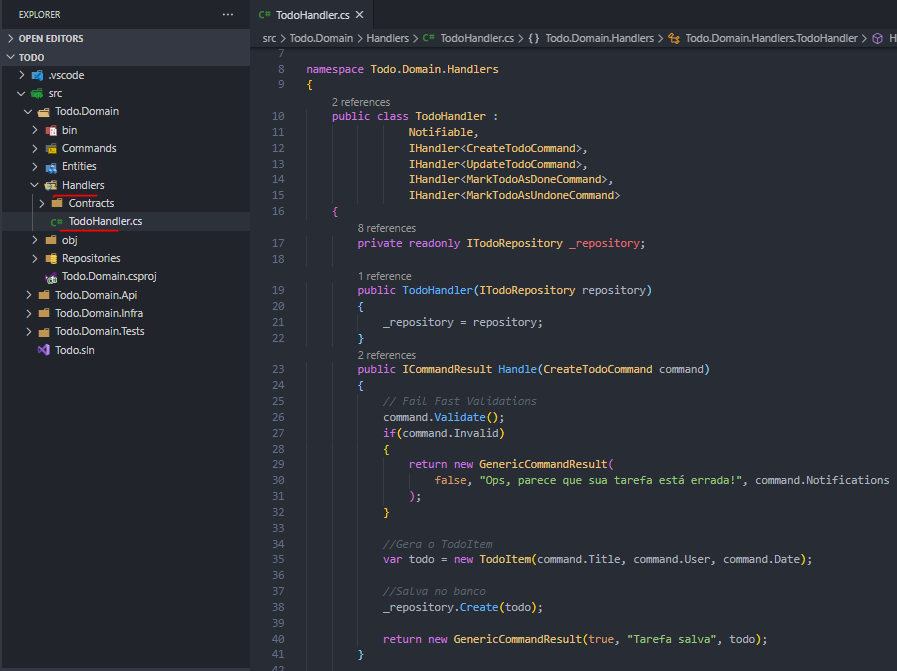
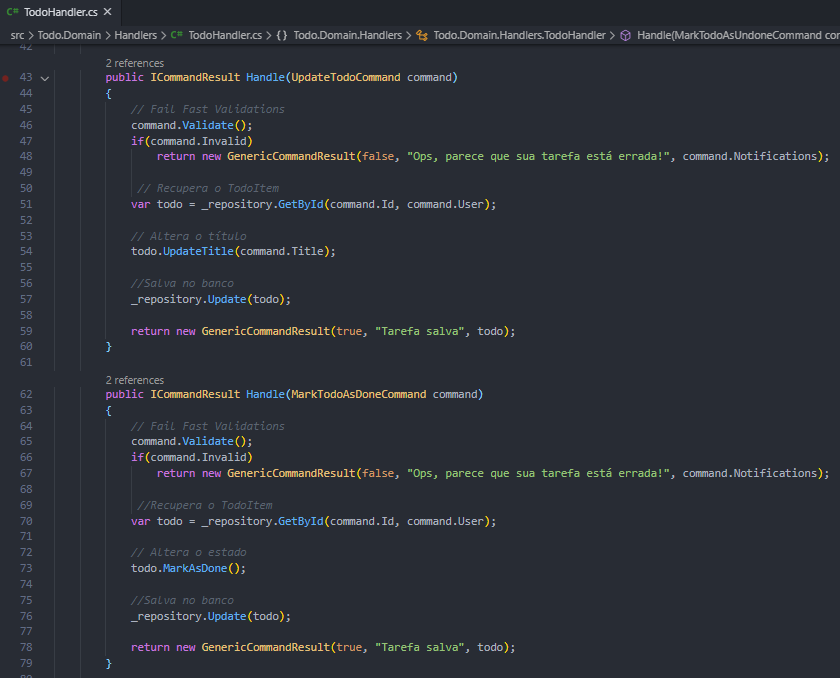
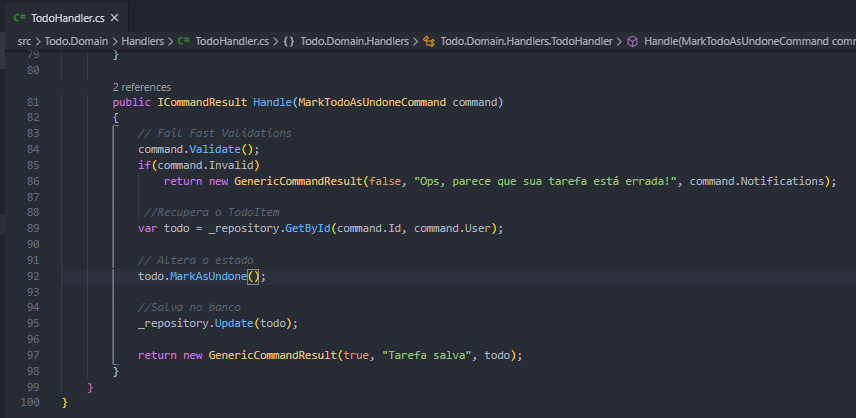


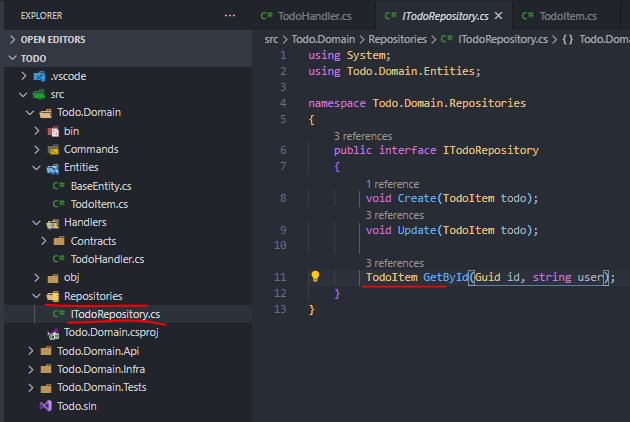
**TESTE DE UNIDADOS**

**COMMAND**

**Terminando o TodoHandler**



**INSTALAR DEPENDENCIA DE BANCO DE DADOS NA MEMORIA**

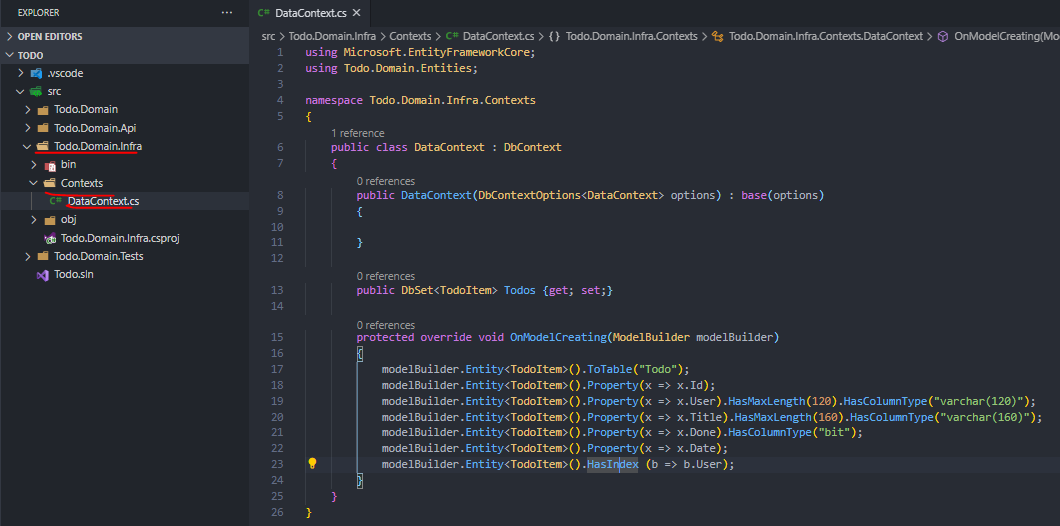
src> cd .\Todo.Domain.Infra\

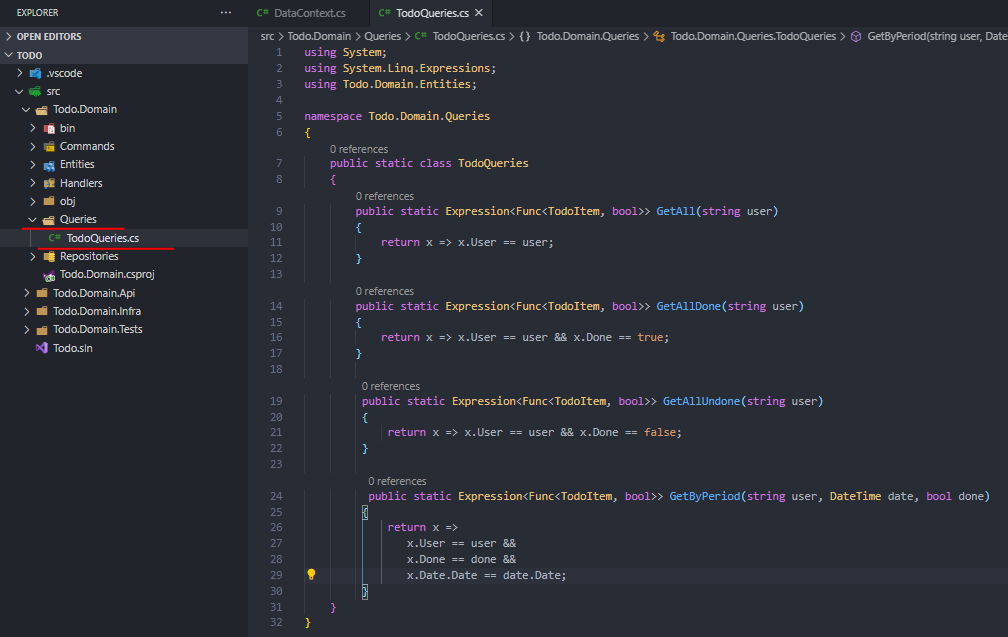
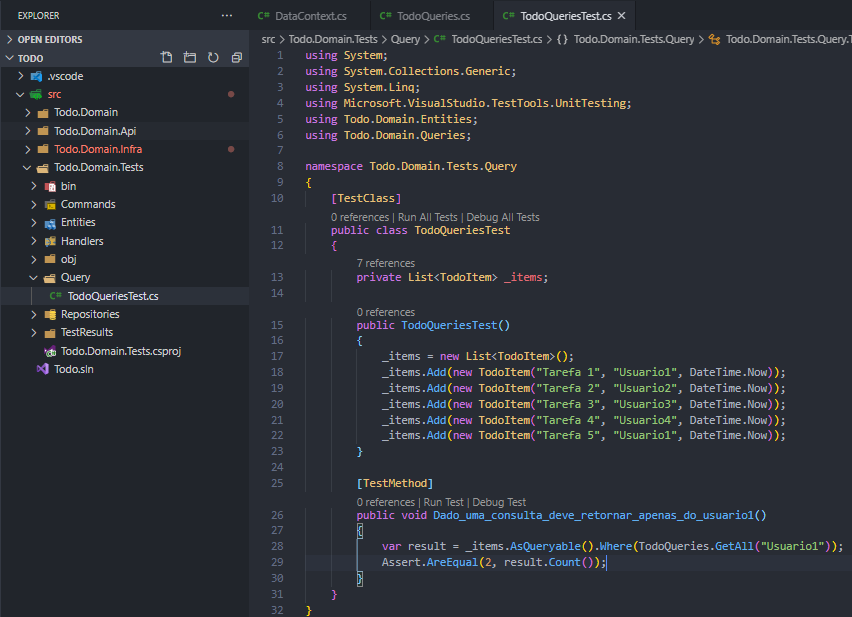
src\Todo.Domain.Infra> dotnet add package Microsoft.EntityFrameworkCore.InMemory

src\Todo.Domain.Infra> dotnet add package Microsoft.EntityFrameworkCore.Design

src\Todo.Domain.Infra> dotnet add package Microsoft.EntityFrameworkCore.Tools

src\Todo.Domain.Infra> dotnet add package Microsoft.EntityFrameworkCore.Relational



**RESOLVENDO DEPENDENCIAS**

src> cd .\Todo.Domain.Api\

src\Todo.Domain.Api> dotnet add package Microsoft.EntityFrameworkCore.InMemory

src\Todo.Domain.Api> dotnet add package Microsoft.EntityFrameworkCore.Design

src\Todo.Domain.Api> dotnet add package Microsoft.EntityFrameworkCore.SqlServer

src\Todo.Domain.Infra> dotnet add package Microsoft.EntityFrameworkCore.SqlServer

**Instalar o EF**

src> dotnet tool uninstall --global dotnet-ef

src> dotnet tool install --global dotnet-ef

**Criar Migrações**

src\Todo.Domain.Infra> dotnet ef migrations add initialCreate --startup-project ..\Todo.Domain.Api\

src\Todo.Domain.Infra> dotnet ef database update --startup-project ..\Todo.Domain.Api\