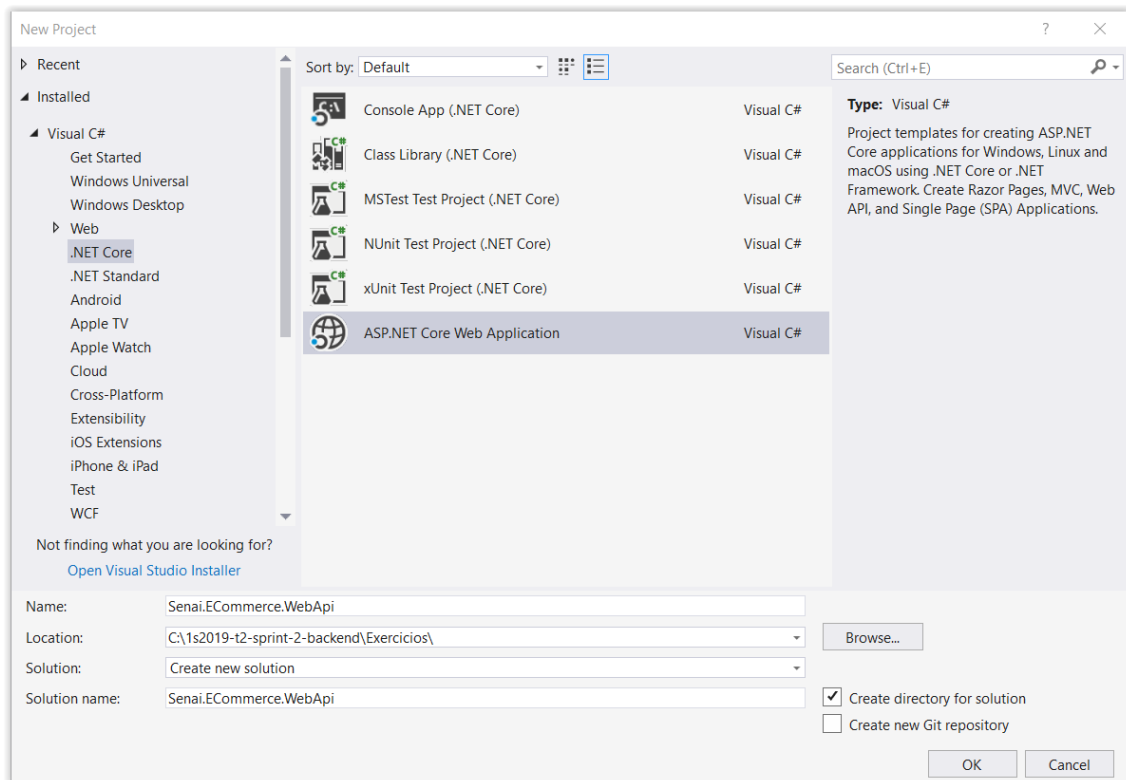
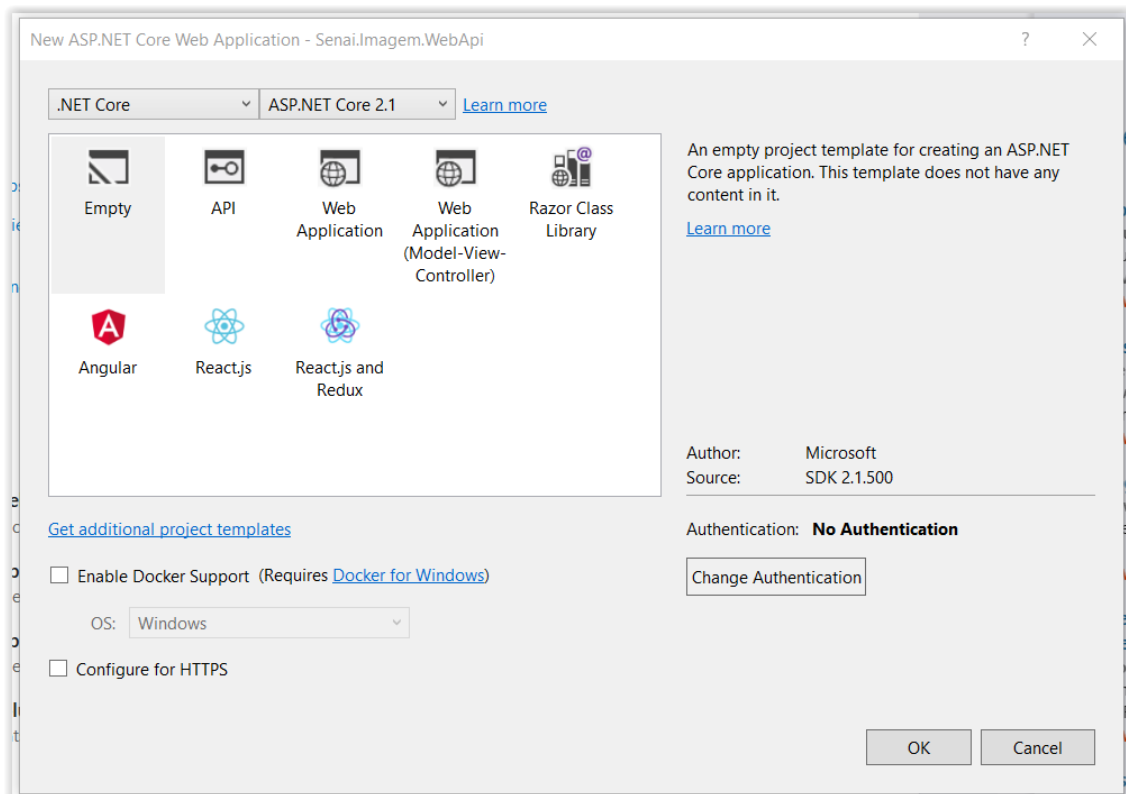


Crie um projeto Asp.Net Core Web Application, escolha a pasta que o mesmo irá ser salvo e defina o nome do mesmo, clique em ok



Escolha a opção Empty e desmarque “Configure for https”



Configure a classe Startup para usar MVC, como iremos trabalhar com imagem e iremos salvar a imagem em uma pasta dentro de nossa aplicação definir UseStaticFiles()

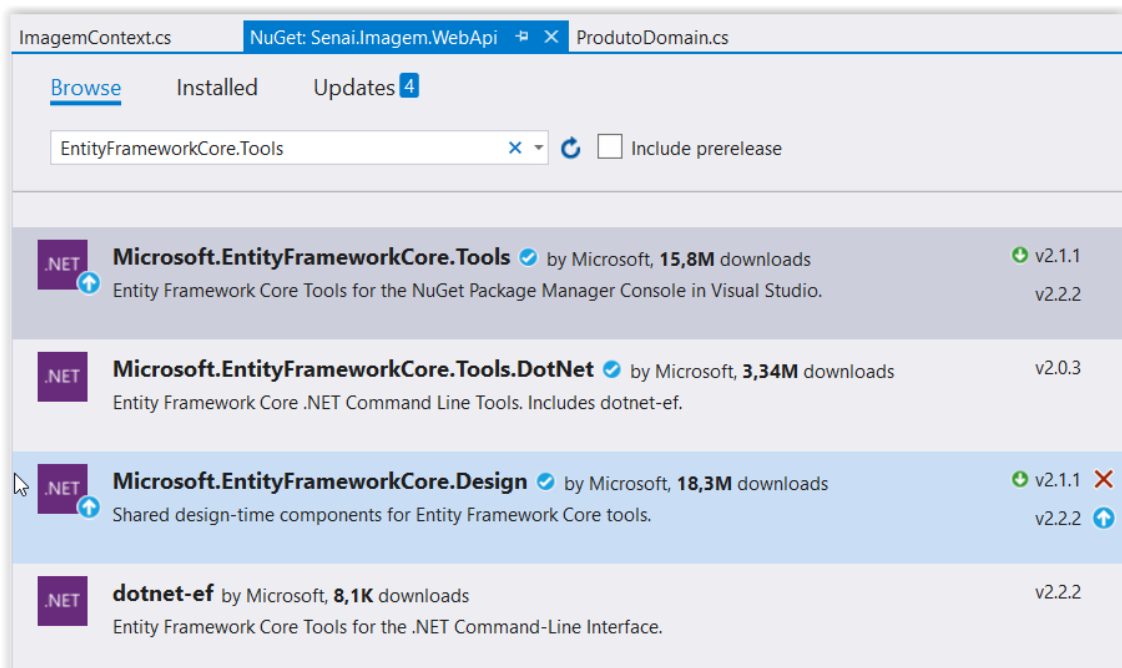
```
public void ConfigureServices(IServiceCollection services)
{
    services.AddMvc().SetCompatibilityVersion(CompatibilityVersion.Version_2_1);
}

// This method gets called by the runtime. Use this method to configure the HTTP request pipeline.
public void Configure(IApplicationBuilder app, IHostingEnvironment env)
{
    if (env.IsDevelopment())
    {
        app.UseDeveloperExceptionPage();
    }

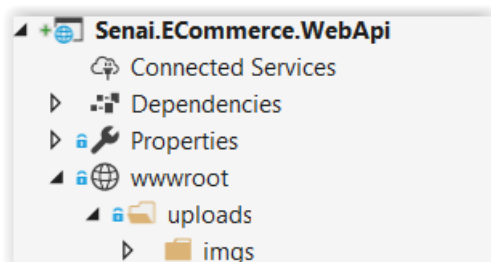
    app.UseStaticFiles();

    app.UseMvc();
}
```

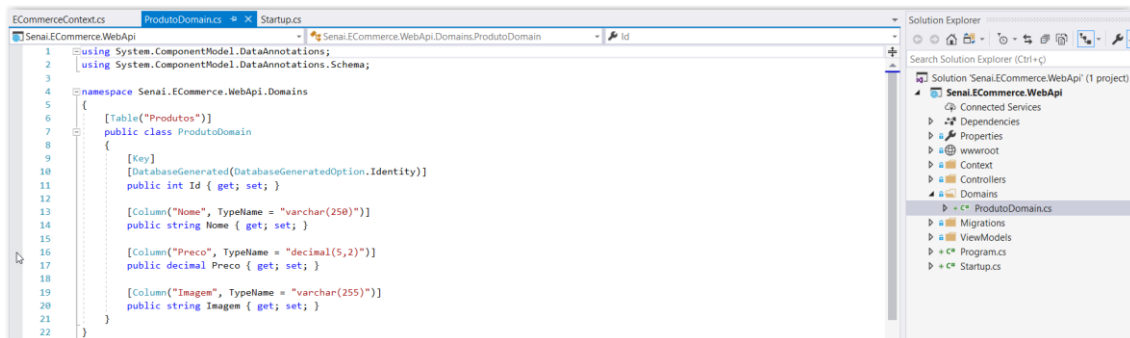
Baixe via nugget os pacotes



Crie um pasta em seu projeto dentro de wwwroot como o nome uploads e dentro dela uma pasta com o nome de imgs



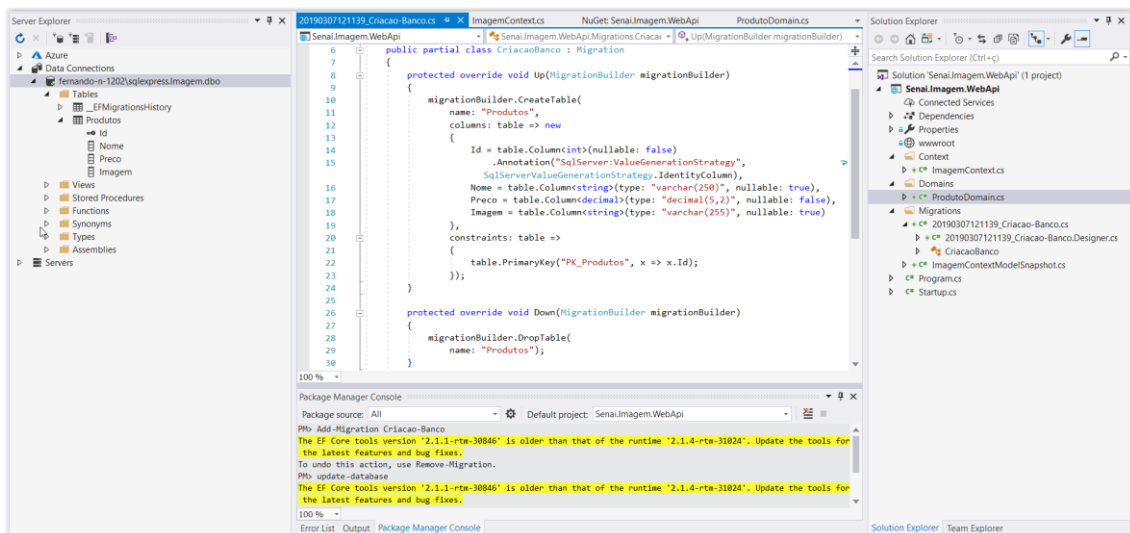
Neste cenário iremos trabalhar com CodeFirst, crie uma pasta Domains e dentro dela uma Classe ProdutoDomain incluindo os DataAnnotation, no Domínio iremos definir a imagem como string pois iremos armazenar o caminho do arquivo neste campo.



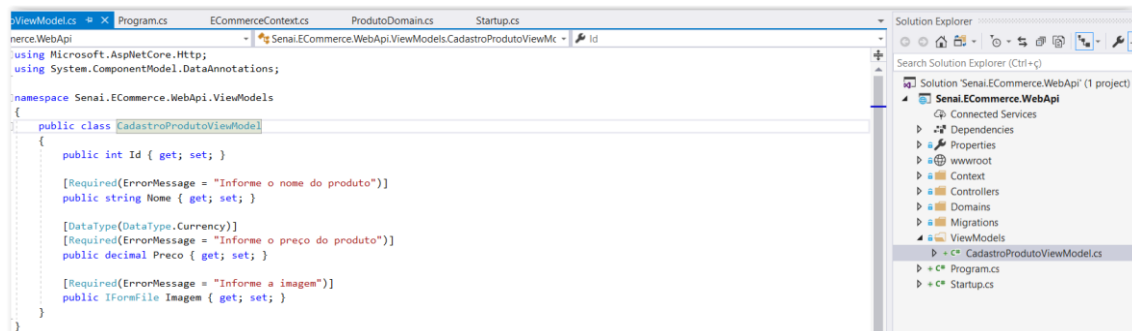
Crie uma pasta Context e crie uma classe ECommerceContext e inclua o DbSet e configure a string de conexão.



Crie o Arquivo de migração e de um update no banco, abra o Package Manager Console e execute os comando Add-Migration Criacao-Banco e Update-Database



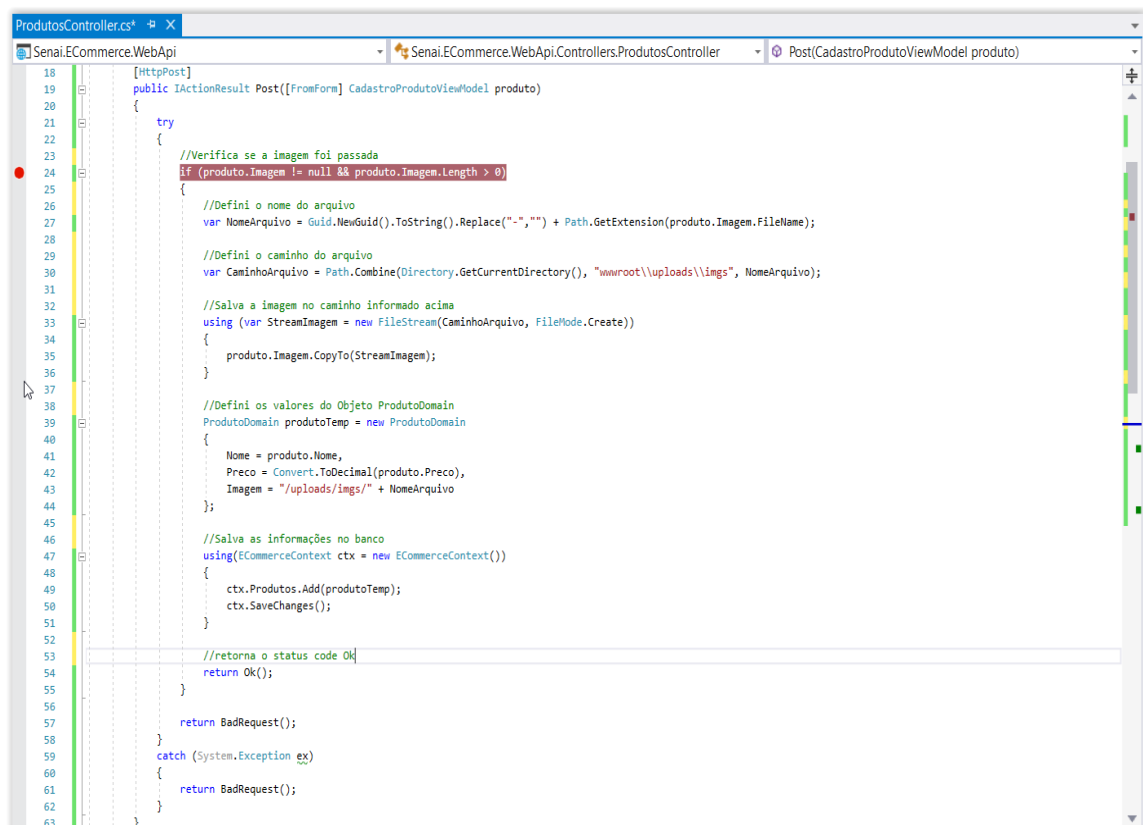
Crie uma pasta ViewModels e crie uma classe CadastroProdutoViewModel, o campo Imagem em nosso ViewModel será do tipo IFormFile pois iremos receber como parâmetro um arquivo de imagem para salvar em uma pasta do projeto e em nosso banco de dados iremos salvar apenas o caminho.



Crie uma pasta Controllers e Adicione um ApiController Empty, de o nome de ProdutosController.

Crie um HttpPost que irá receber um CadastroProdutoViewModel com o arquivo da imagem, irá salvar a mesma no hd e irá salvar o caminho da imagem no banco como na figura abaixo.

Para receber a imagem como parâmetro incluímos [FromForm]



Abra o Postman e crie uma nova requisição.

POST http://localhost:5000/api/Produtos

Send Save

Params Authorization Headers Body Pre-request Script Tests Cookies Code Comments (0)

none form-data x-www-form-urlencoded raw binary

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	Nome	MacBook Air MQD32BZ/A com Intel Core i5 Du...			
<input checked="" type="checkbox"/>	Imagem	ext.jpg X			
<input checked="" type="checkbox"/>	Preco	574,74			
	Key	Value	Description		

Body Cookies Headers (3) Test Results Status: 200 OK Time: 11338 ms Size: 92 B

No tipo de campo para Imagem escolha file e selecione a mesma do seu computador, atente-se para a opção form-data.

Verifique no seu banco de dados se o produto foi salvo.

```
1 SELECT TOP (1000) [Id]
2     , [Nome]
3     , [Preco]
4     , [Imagem]
5 FROM [ECommerce].[dbo].[Produtos]
6
```

RESULTS				
	Id	Nome	Preco	Imagem
1	1	MacBook Air MQD32BZ/A com Intel Cor...	574,74	/uploads/imgs/fd44b3128d4c4434bc033f00d74bafd5.jpg

Para visualizar a imagem será a url da sua api mais o caminho da imagem, na figura acima a url da imagem será http://localhost:5000/uploads/imgs/fd44b3128d4c4434bc033f00d74bafd5.jpg