**DESAFIO FIAP**

Tecnologias usadas: ASP.NET MVC CORE, ENTITY FRAMEWORK CORE,SQL SERVER, C#;

**Caso houver algum problema para rodar a aplicação, ela está disponível em:**

<http://danilodev.somee.com/>

VIDEOS SOBRE A ARQUITEUTRA E APLICAÇÃO:

Arquitetura

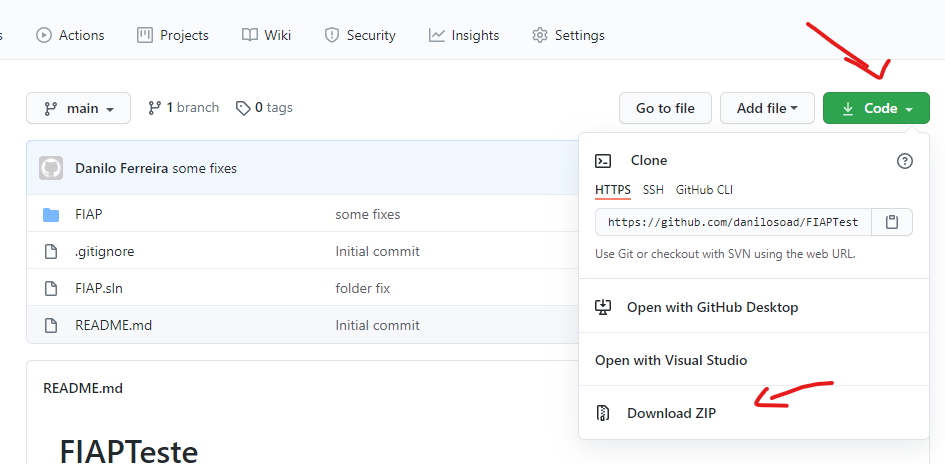
<https://www.youtube.com/watch?v=n8K3NvWCFiA>

Aplicação rodando:

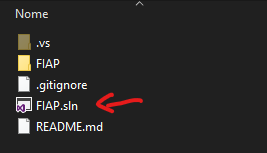
<https://www.youtube.com/watch?v=BphZk-F1Ru8>

Como rodar a aplicação:

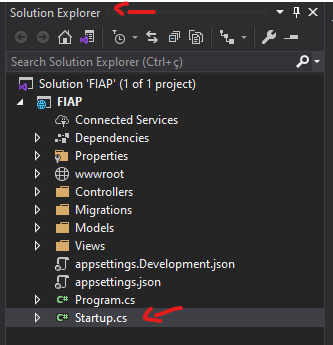
1. Acessar o repositório do **GitHub** ( <https://github.com/danilosoad/FIAPTeste> ) e baixar o ZIP da aplicação conforme imagem abaixo.



1. Extrair os arquivos e executar o arquivo SLN ou abrir o arquivo pelo Visual Studio 19



1. Acessar o Solution Explorer e abrir a classe **Startup.cs**

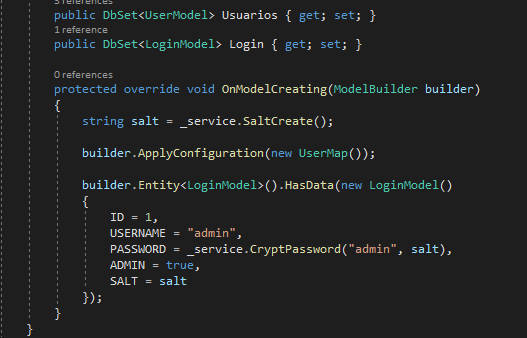


Procurar pelo método **ConfigureServices**, localizar o **services.AddDbContext** , **comentar** a linha que aponta para o banco web da aplicação e **descomentar** a que aponta para o banco local.

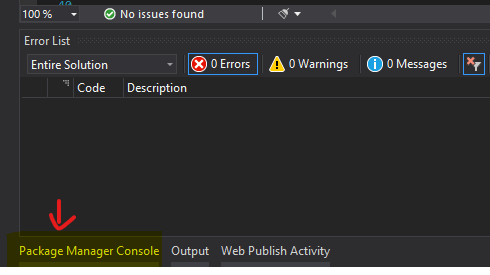


Após isso alterar a string de conexão para o servidor(**Data Source**) e o banco (**Initial Catalog**) que será utilizado para o teste. 

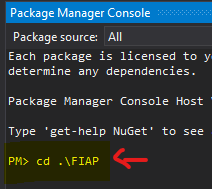
PS: Apesar de haver um script SQL para criação das tabelas, foi utilizado a abordagem CODE FIRST, dessa maneira eu recomendo que seja criada as tabelas pela aplicação pois já terá uma carga inicial para o login do administrador.



1. Criar as tabelas através do EF CORE
   1. Abra o Packager Manager Console na parte inferior esquerda do VisualStudio

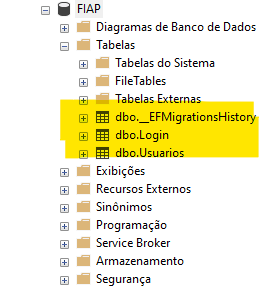


* 1. Entre na pasta do projeto caso não esteja, utilizando o comando cd

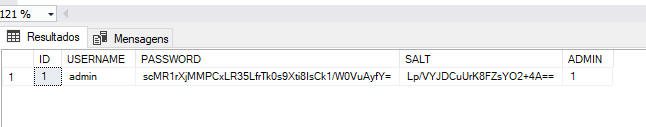


* 1. Execute o comando **dotnet ef database update** para criar as tabelas Login, Usuarios e a de migrações do EFCORE.





A tabela Login já possuirá uma carga Inicial para o uso na aplicação



Feito isso a aplicação já pode ser utilizada, para o acesso a tela do administrador o login: **admin** e a senha : **admin**

