

Projeto To-Do List em ASP.NET Core MVC

Realizado por:

António Elói – N° 230000932

Cristian Grigorita – N° 230000287

Guilherme Barbosa – N° 230000002

Docente:

Ana Catarina Cruz

Santarém, Santarém

10 de janeiro de 2025

Neste trabalho, desenvolvemos uma aplicação web feita em C#, HTML5, CSS, SQLServer, que utiliza a framework ASP.NET para a construção de web apps acerca de uma lista desenvolvida conforme a matéria dada nas aulas da disciplina de Tecnologias de Internet lecionada pela Professora Ana Catarina Cruz.

O projeto se trata de uma To-Do List que possui utilizadores divididos em Admin, Gestores e Clientes, o utilizador Admin é definido e criado ao rodar o programa caso ainda não exista.

Credenciais de admin para seu acesso:

Email: **admin@geral.com**

Password: **admin12345**

Estas credenciais encontram-se na classe Program.cs, na solução do projeto, onde o admin é definido e criado.

Credenciais dos gestores pré-criados:

Email: **teste1@gestor.com** | Password: **teste123**

Email: **teste2@gestor.com** | Password: **teste123**

Email: **teste3@gestor.com** | Password: **teste123**

Credenciais dos clientes pré-criados:

Email: **teste1@cliente.com** | Password: **teste123**

Email: **teste2@cliente.com** | Password: **teste123**

Email: **teste3@cliente.com** | Password: **teste123**

Portanto, todo utilizador que se registar no site será definido como Cliente, podendo o Admin passar um Cliente para Gestor, se for de sua vontade.

Infelizmente, não conseguimos conectar nosso projeto ao Azure, mais para frente iremos mostrar prints do nosso server feito nele e o erro que obtivemos.

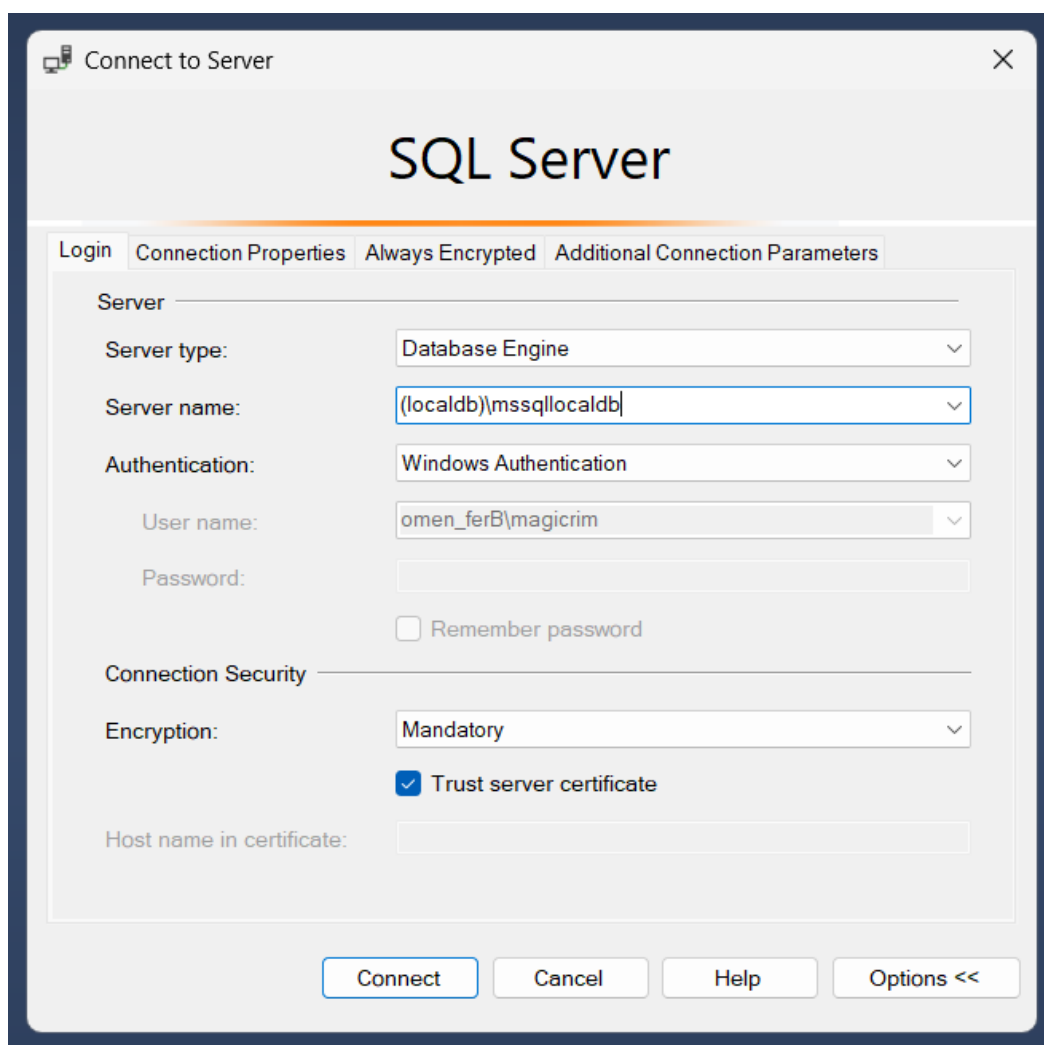
Por conta disso, optamos por fazer um backup da base de dados local para que seja possível aceder aos dados já preenchidos por padrão.

Para se conectar à base de dados, deve seguir os seguintes passos:

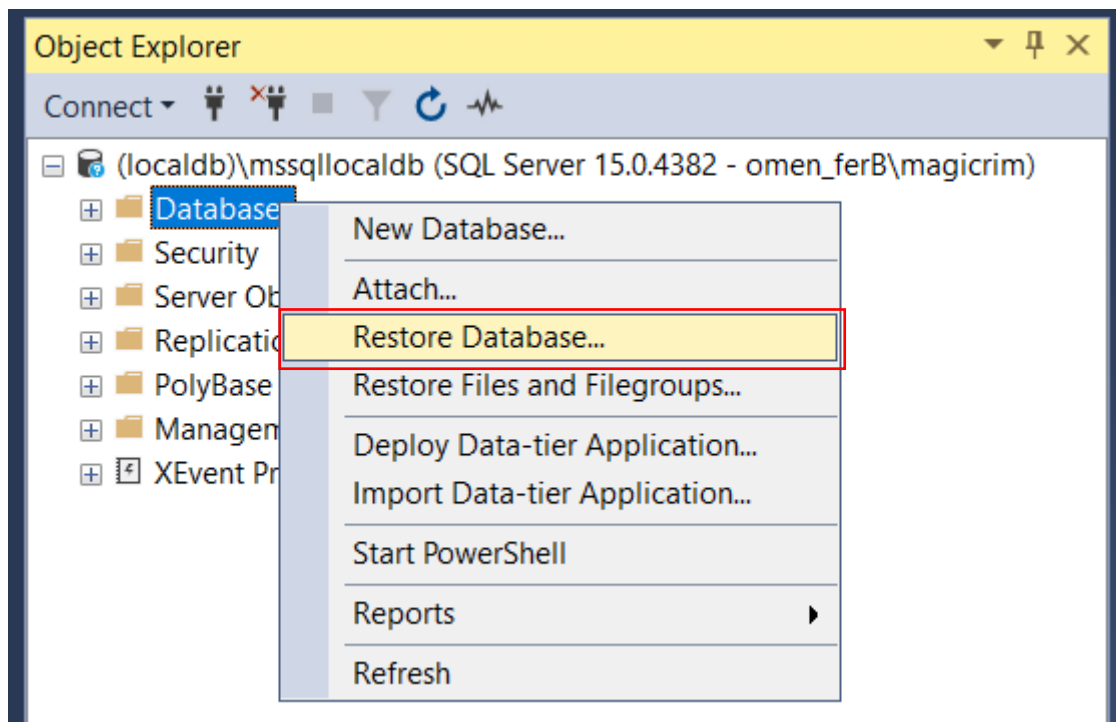
1. Após fazer extrair o projeto do ficheiro zipado ou clonar o repositório pelo github, localizar o ficheiro de backup da base de dados (ToDoDBBackup.bak):

Nome	Data de modificação	Tipo	Tamanho
.vs	10/01/2025 21:49	Pasta de arquivos	
ToDo	10/01/2025 21:48	Pasta de arquivos	
.gitattributes	10/01/2025 21:45	Arquivo Fonte Git ...	3 KB
.gitignore	10/01/2025 21:45	Arquivo Fonte Git I...	7 KB
README.md	10/01/2025 21:45	Arquivo MD	2 KB
ToDo.sln	10/01/2025 21:45	Visual Studio Solut...	2 KB
ToDo	10/01/2025 21:45	SQL Text File	4 KB
ToDoDBBackup.bak	10/01/2025 21:45	Arquivo BAK	4 055 KB

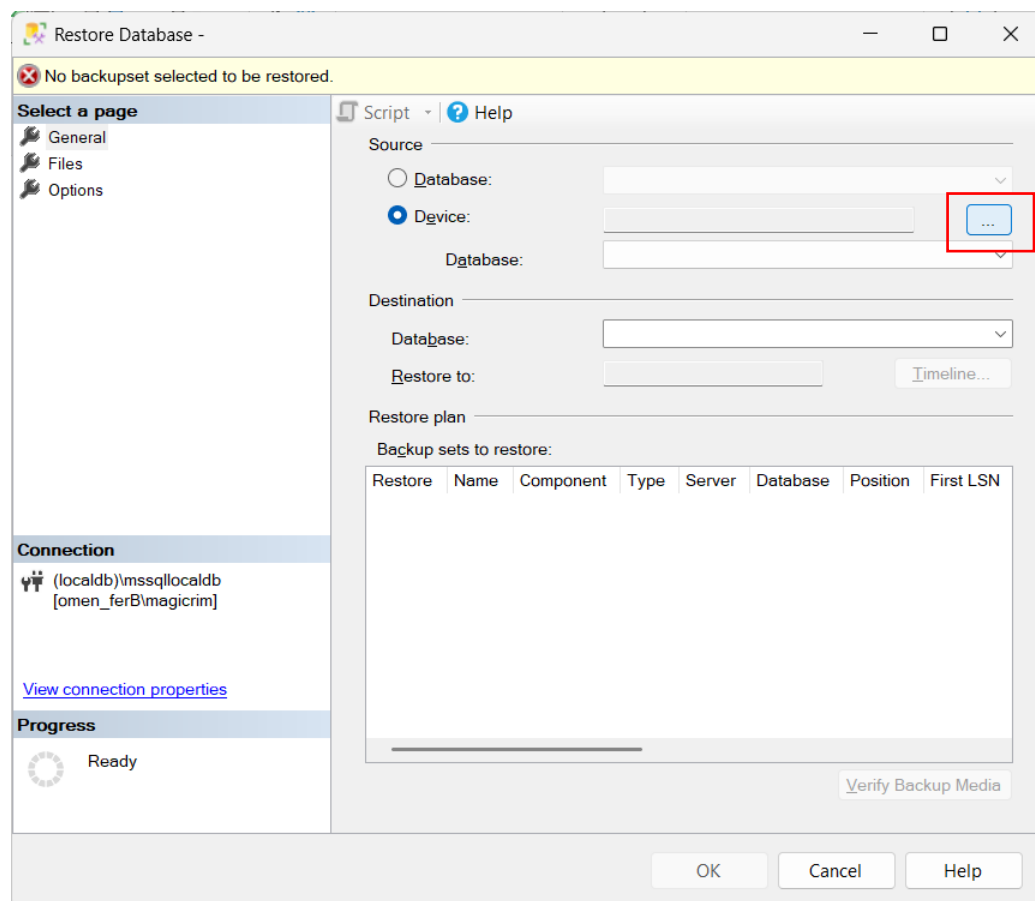
2. Abrir o SQL Server Management Studio (SSMS) e conectar-se ao servidor de banco de dados '(localdb)\mssqllocaldb':



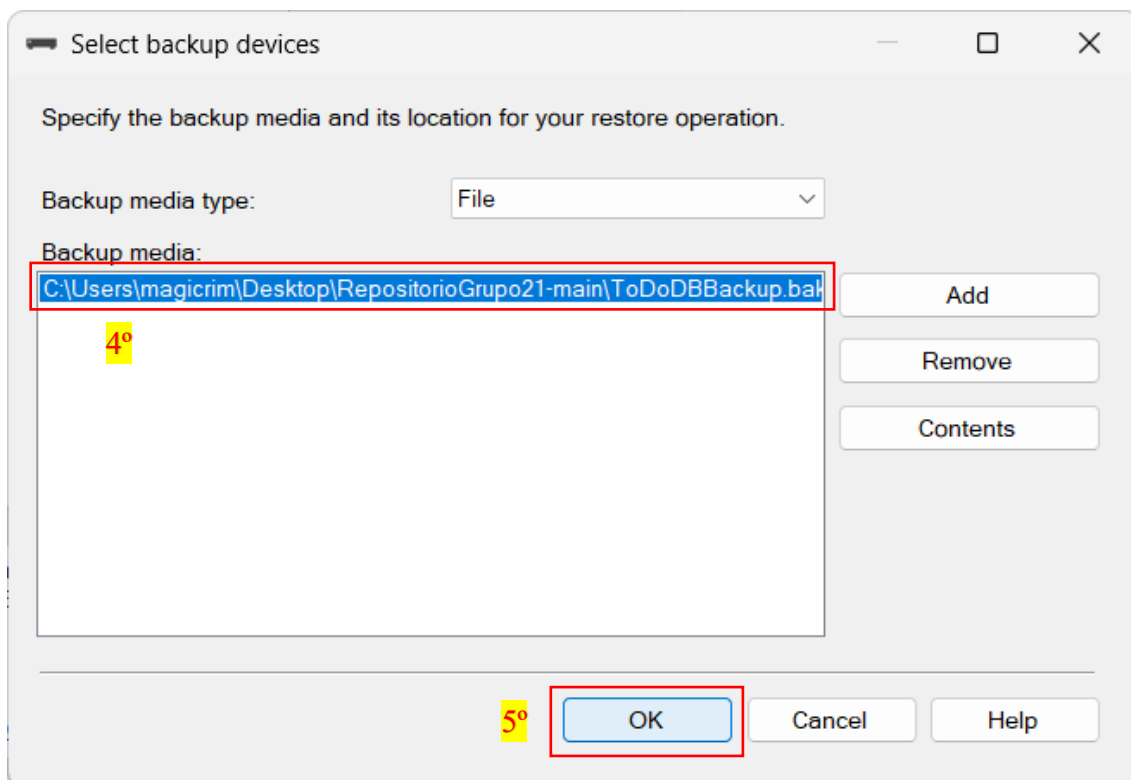
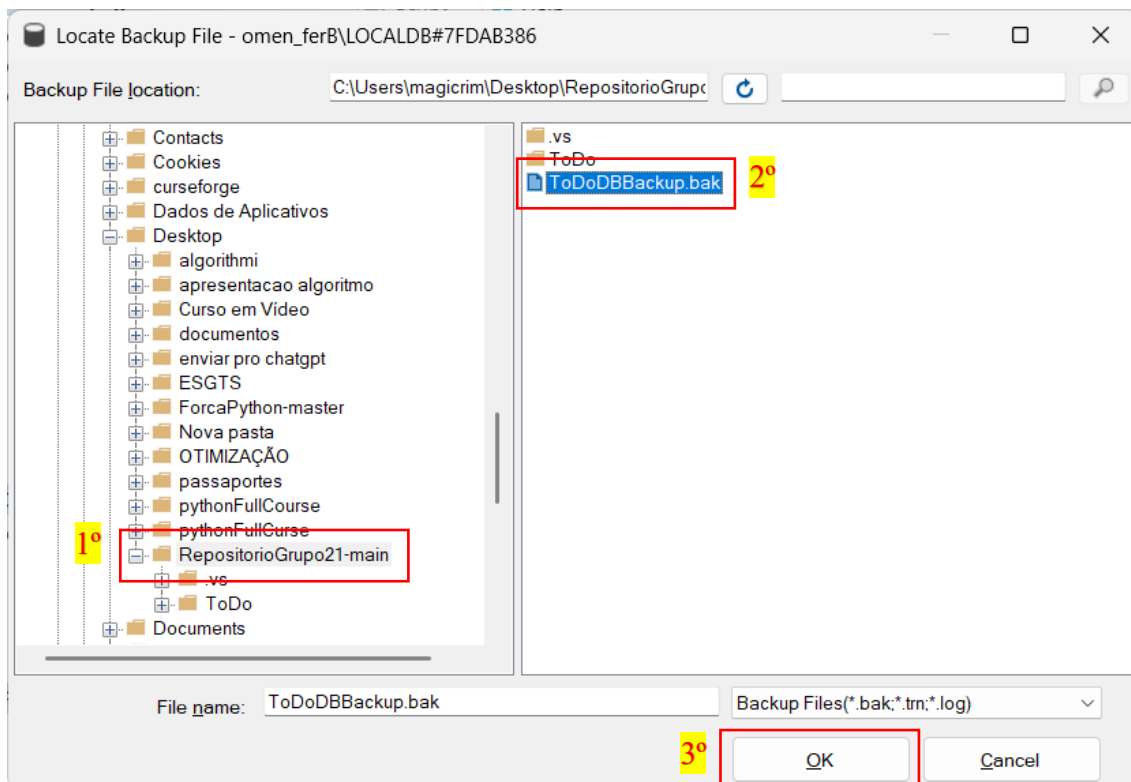
3. No Object Explorer, clicar com o botão direito no nó Databases e seleccionar Restore Database...:



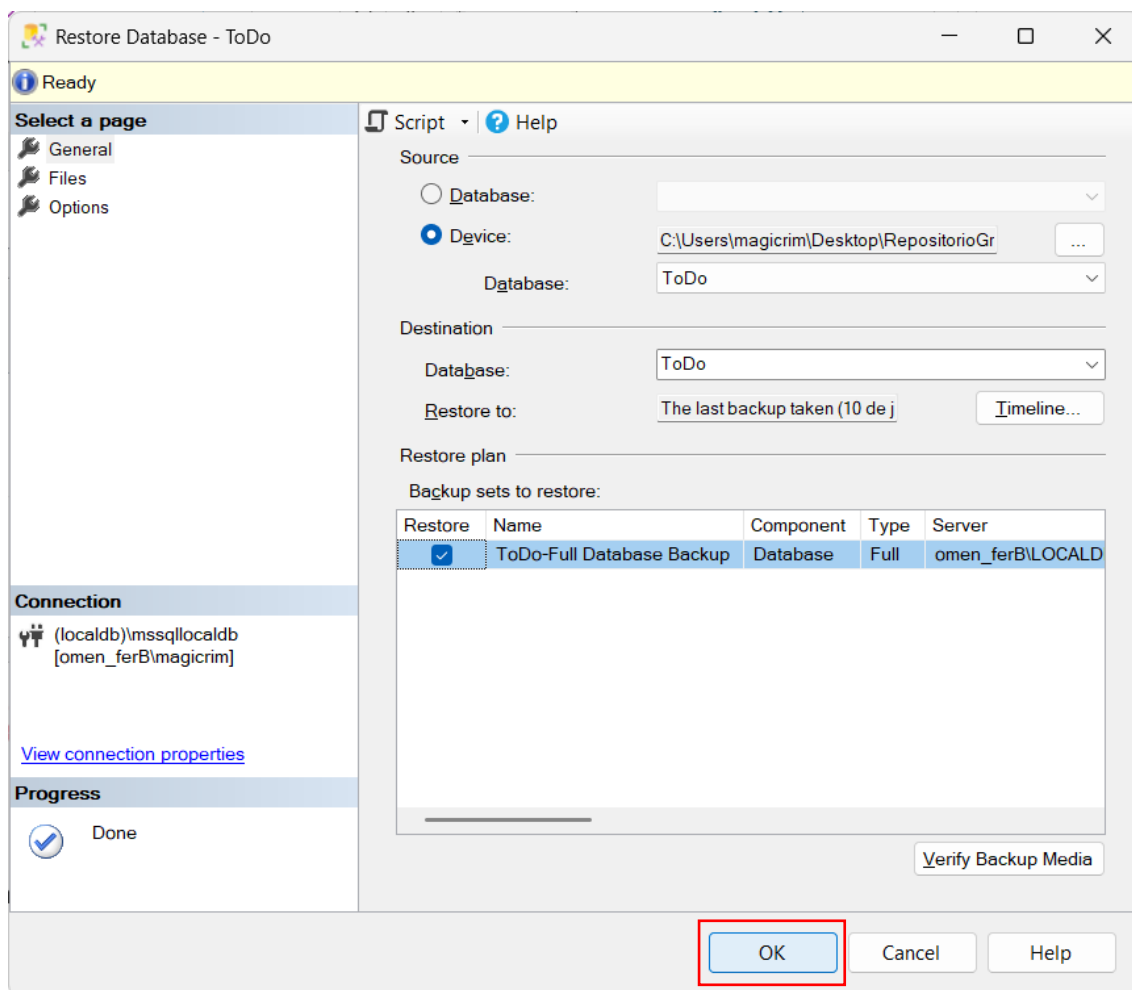
4. Na janela de restauração, seleccionar 'Device' e clicar no botão '...' para seleccionar o arquivo de backup:



5. Clicar em 'Add' e navegar até o local onde o arquivo de backup está salvo (por exemplo, RepositorioGrupo21-main\ToDoDBBackup.bak). Após isso, selecionar o arquivo de backup e clicar em OK.



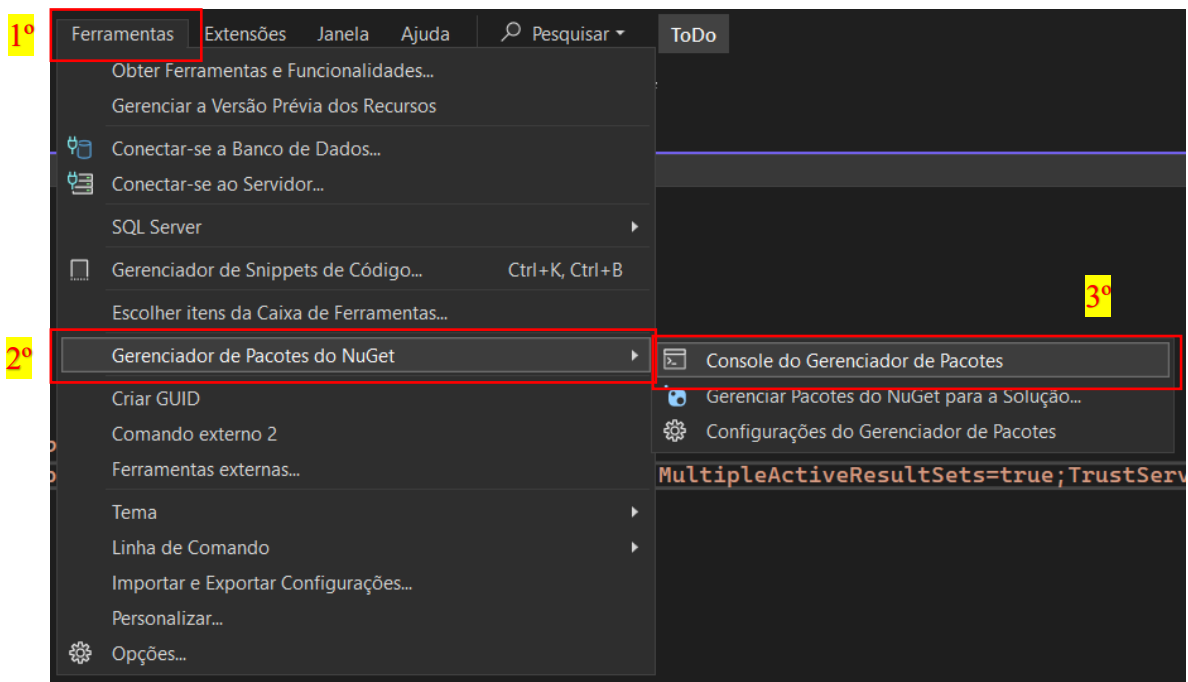
- Configurar as opções de restauração conforme necessário e clicar em OK para iniciar a restauração:



- Após isso, a database será criada no Object Explorer com o nome 'ToDo'. Agora no projeto, é necessário verificar se a string de conexão no arquivo appsettings.json e/ou Program.cs está configurada corretamente para apontar para o servidor de base de dados que definimos anteriormente. Já o definimos por padrão no ficheiro "appsettings.json", então não é necessário fazer nenhuma modificação.

```
"ConnectionStrings": {
  "ToDoConnection": "Server=localhost;Database=ToDo;Trusted_Conn
  "ToDoContext": "Server=(localdb)\mssqllocaldb;Database=ToDo;T
},
"AllowedHosts": "*"
}
```

8. Abrir o projeto no Visual Studio. Na barra superior clicar em Ferramentas > Gerenciador de Pacotes do NuGet > Console do Gerenciador de Pacotes:



9. Na janela do console, executar o comando 'Update-Database':

```
PM> Update-Database
Build started...
Build succeeded.
Microsoft.EntityFrameworkCore.Database.Command[20101]
    Executed DbCommand (15ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    SELECT 1
Microsoft.EntityFrameworkCore.Database.Command[20101]
    Executed DbCommand (13ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    SELECT OBJECT_ID(N'[__EFMigrationsHistory']);
Microsoft.EntityFrameworkCore.Database.Command[20101]
    Executed DbCommand (1ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    SELECT 1
Microsoft.EntityFrameworkCore.Database.Command[20101]
    Executed DbCommand (0ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    SELECT OBJECT_ID(N'[__EFMigrationsHistory']);
Microsoft.EntityFrameworkCore.Database.Command[20101]
    Executed DbCommand (3ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
    SELECT [MigrationId], [ProductVersion]
    FROM [__EFMigrationsHistory]
    ORDER BY [MigrationId];
Microsoft.EntityFrameworkCore.Migrations[20405]
    No migrations were applied. The database is already up to date.
No migrations were applied. The database is already up to date.
Done.
```

10. Após isso, compilar e executar o projeto normalmente que a base de dados já deve estar atualizada.

Prints do Azure (tentamos de diversas maneiras fazer funcionar, mas não conseguimos:

The screenshot shows the Microsoft Azure portal interface. The top navigation bar includes the Microsoft Azure logo, a search bar, and user information. The main content area displays the configuration for a SQL Server resource named 'dbs-esgts-tigrupo21'. The left sidebar shows navigation options like 'Visão geral', 'Log de atividade', 'IAM (Controle de acesso)', 'Marcações', 'Início rápido', and 'Diagnosticar e resolver problemas'. The main panel shows the 'Fundamentos' (Basics) tab with fields for 'Grupo de recursos' (Resource group), 'Status', 'Local' (Location), 'Assinatura' (Subscription), 'ID da Assinatura' (Subscription ID), and 'Marcações' (Tags). On the right, there are fields for 'Administrador do servidor' (Server administrator), 'Rede' (Network), 'Administrador do Micros...' (Microsoft administrator), and 'Nome do servidor' (Server name).

The screenshot shows the 'Connect to Server' dialog box in SQL Server Enterprise Manager. The 'Server' tab is selected, showing the following configuration:

- Server type: Database Engine
- Server name: dbs-esgts-tigrupo21.database.windows.net
- Authentication: SQL Server Authentication
- Login: ti-grupo21
- Password: (masked with asterisks)
- ☒ Remember password
- Connection Security: Encryption: Mandatory
- ☐ Trust server certificate
- Host name in certificate: (empty field)

At the bottom, there are buttons for 'Connect', 'Cancel', 'Help', and 'Options <<'.

The screenshot shows an error message dialog box titled 'Connect to Server'. The message states: 'Cannot connect to dbs-esgts-tigrupo21.database.windows.net. Additional information: Login failed for user 'ti-grupo21'. (Microsoft SQL Server, Error: 18456)'. At the bottom, there are buttons for 'Help', 'Copy message', 'Show details', and 'OK'.