Todo

Version 0.4 – 28 October 2023 | Naji El Kotob

Contents

[Source Code 2](#_Toc149229455)

[GitHub 2](#_Toc149229456)

[Database 2](#_Toc149229457)

[SQL Server Database 2](#_Toc149229458)

[API 2](#_Toc149229459)

[Configure ConnectionString 2](#_Toc149229460)

[Run the API and Test with Swagger 2](#_Toc149229461)

[Call a web API 4](#_Toc149229462)

[Call a web API from ASP.NET Core Blazor 4](#_Toc149229463)

[Call a web API from .NET MAUI 4](#_Toc149229464)

## Source Code

### GitHub

1. <https://github.com/NajiElKotob/Blazor-TodoList>

## Database

### SQL Server Database

1. Create a new Database and name it “TodoList”.
2. Create a new table named “Todos” (Script below).

USE TodoList

Go

CREATE TABLE [dbo].[Todos] (

[Id] INT IDENTITY (1, 1) NOT NULL,

[Title] NVARCHAR (50) NOT NULL,

[IsDone] BIT NULL,

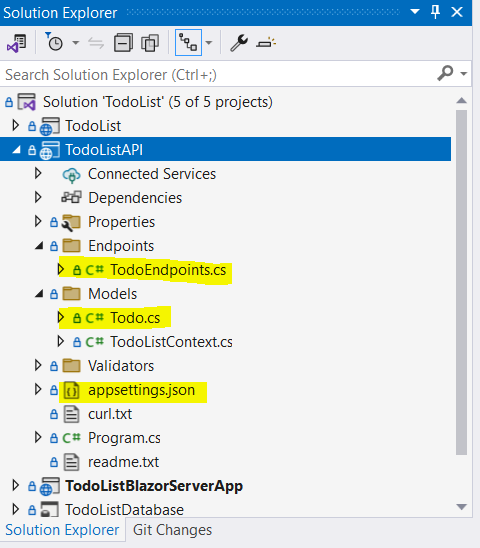
PRIMARY KEY CLUSTERED ([Id] ASC)

);

## API

### Configure ConnectionString

1. Open TodoListAPI > appsettings.json and update the connection string accordingly. Mainly, check the server name and database name.

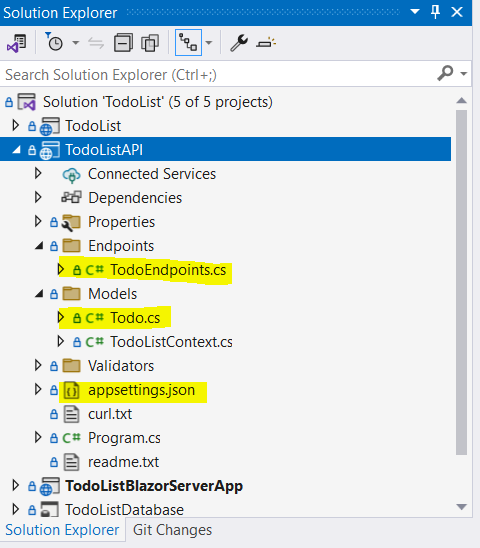


### Run the API and Test with Swagger

1. Set `TodoListAPI` as the startup application (Right-click > Set as Startup Project).
2. Launch the application.
3. Test the available API methods using Swagger.
4. Review the result messages.
5. Verify the database records.

API Code

Explore the code located in `Models > Todo` and `Endpoints > TodoEndpoints`.



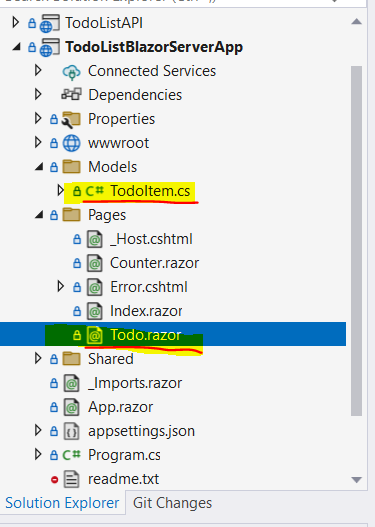
#### References

* Minimal APIs quick reference <https://learn.microsoft.com/en-us/aspnet/core/fundamentals/minimal-apis?view=aspnetcore-7.0>

## Call a web API

### Call a web API from ASP.NET Core Blazor

Explore the code located in `Models > TodoItem` and `Pages > Todo.razor`.



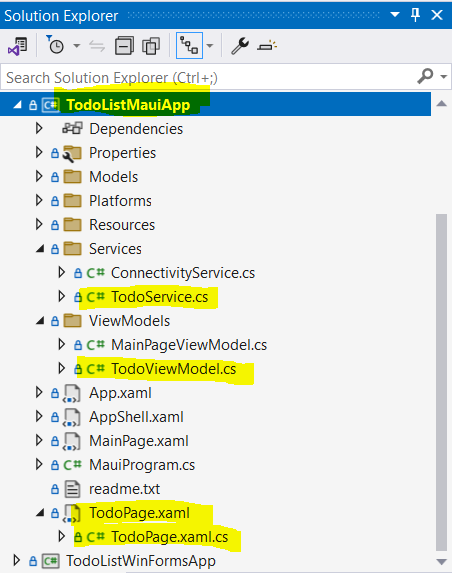
Tip! [Set multiple startup projects - Visual Studio (Windows)](https://learn.microsoft.com/en-us/visualstudio/ide/how-to-set-multiple-startup-projects?view=vs-2022)

#### References

* Call a web API from ASP.NET Core Blazor <https://learn.microsoft.com/en-us/aspnet/core/blazor/call-web-api?view=aspnetcore-7.0&pivots=server>

### Call a web API from .NET MAUI

Explore the highlighted files (screenshot below).

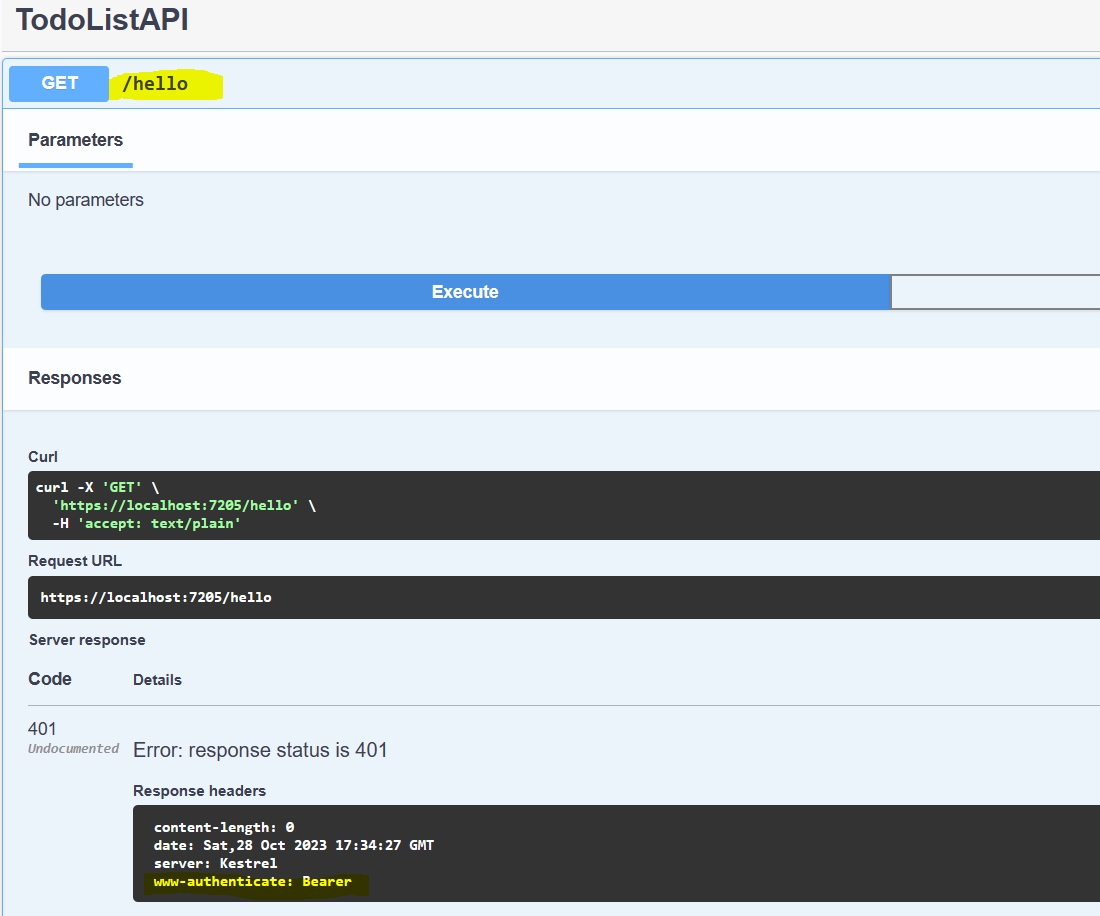


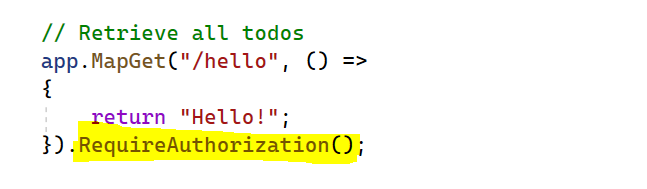
#### References

* Consume a REST-based web service <https://learn.microsoft.com/en-us/dotnet/maui/data-cloud/rest>
* Consume REST web services in .NET MAUI apps <https://learn.microsoft.com/en-us/training/modules/consume-rest-services-maui>
* REST Web Requests with Data Caching with ASP.NET Core Web API | Xamarin & .NET MAUI 101 <https://www.youtube.com/watch?v=a37qBMt0V9w>
  + Source Code <https://github.com/jamesmontemagno/mycoffeeapp?WT.mc_id=friends-0000-jamont>

## Add JWT to a web API

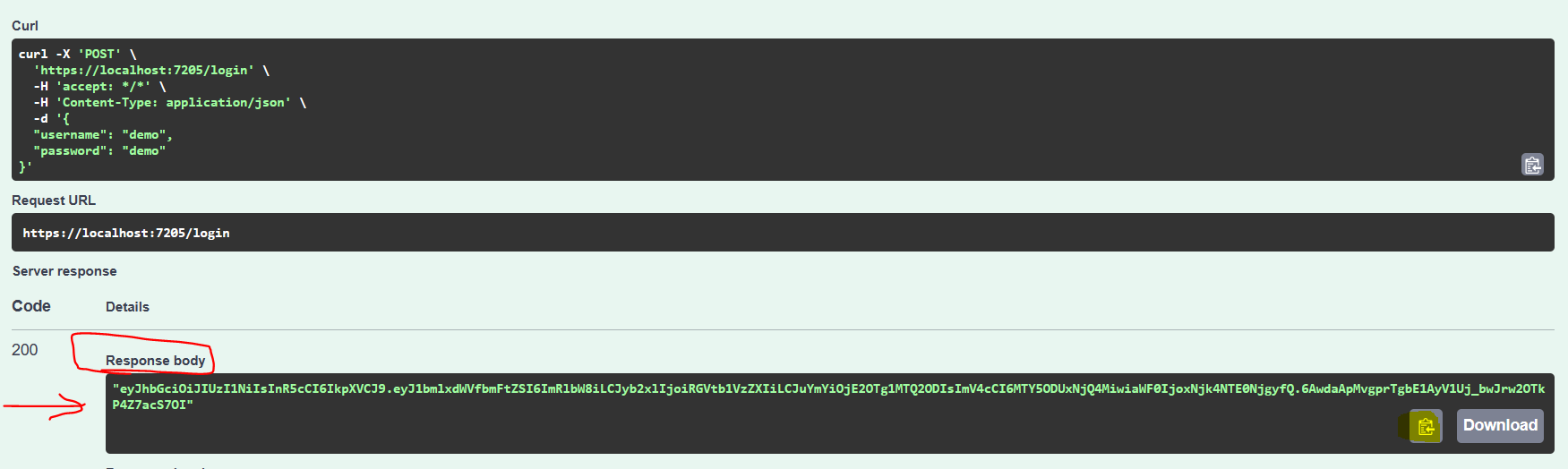
1. Try to execute Hello! Yep, it’s secure.



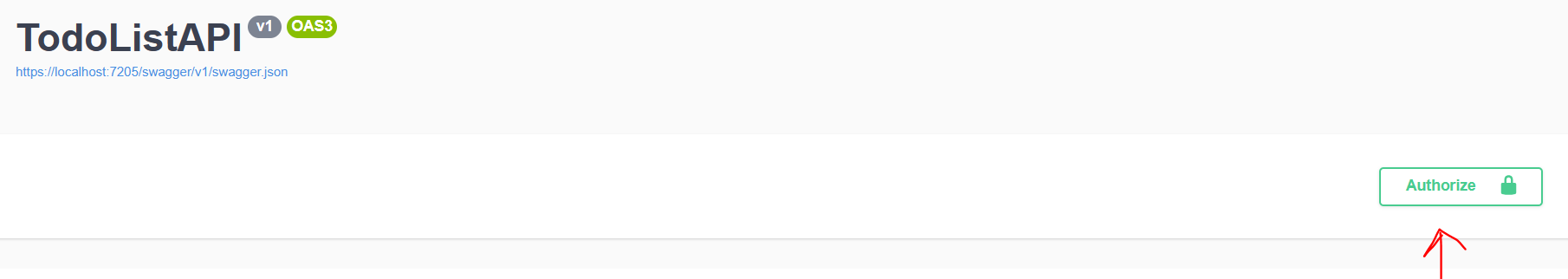


1. Now, login with username: demo/password: demo. Copy the token.





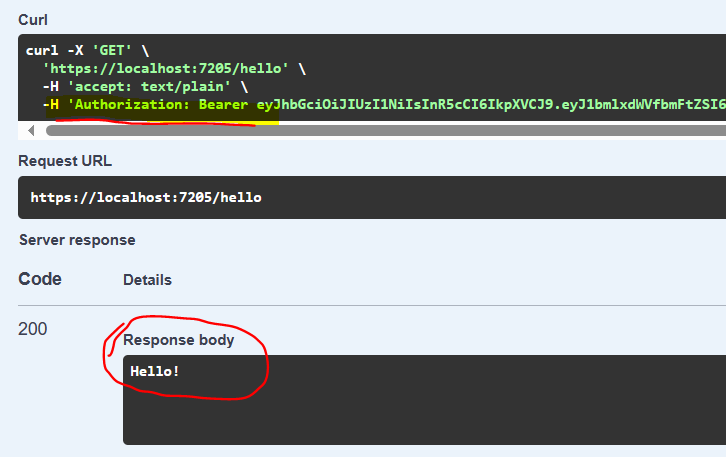
1. Add a prefix “Bearer “ to the token and paste in the “Authorize”



*Example:*

Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1bmlxdWVfbmFtZSI6ImRlbW8iLCJyb2xlIjoiRGVtb1VzZXIiLCJuYmYiOjE2OTg1MTU2NjgsImV4cCI6MTY5ODUxNzQ2OCwiaWF0IjoxNjk4NTE1NjY4LCJpc3MiOiJodHRwczovL2F1dGgudG9kby5sb2NhbCIsImF1ZCI6IlRvZG9BcHBzIn0.h9CCFamQeJwpTk6mP3YyEUaYiN6bdQ78Y0Jcz4I4rZo

1. Execute Hello again, it should work



Happy Programming 😎