Project Manager web API

OVERVIEW

|  |  |  |  |
| --- | --- | --- | --- |
| API | Description | Request body | Response body |
| GET /api/todoitems | Get all to-do items | None | Array of to-do items |
| Create /api/todoitems | Add a new item | To-do item | To-do item |
| Edit /api/todoitems/{id} | Edit an existing item | To-do item | To-do item |
| Delete /api/todoitems/{id} | Delete an item | To-do item | To-do item |

Prerequisites

The Entity Framework provides a great ORM solution that allows you to automatically associate regular C# language classes with tables in your database. Entity Framework Core is primarily aimed at working with the MS SQL Server DBMS, but it also supports a number of other DBMSs. In this case, we will work with databases in MS SQL Server.

It is also worth noting that here we will use Entity Framework Core, a cross-platform solution based on .NET Core, which differs from previous versions, for example, from Entity Framework 6.

To work with the Entity Framework, first create a new ASP.NET Core project using the Web Application template (Model-View-Controller).

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описаниеTo interact with MS SQL Server through the Entity Framework, the Microsoft.EntityFrameworkCore.SqlServer package is required. By default, it is not in the project, so you need to add it, for example, through the Nuget package manager:

That's all we need to work with the project