**ЛАБОРАТОРНА РОБОТА № 1**

**Тема:** **:** Вступ до ASP.NET Core

***Мета роботи:*** ознайомитися з основними принципами роботи .NET, навчитися налаштовувати середовище розробки та встановлювати необхідні компоненти, набути навичок створення рішень та проектів різних типів, набути навичок обробки запитів з використанням middleware.

**Хід роботи:**

# Завдання 1. Встановлення інтегрованого середовища розробки (IDE) та необхідних компонентів

Since I already had Visual Studio 2022 installed and packages configured for ASP.Net and web development (Fig. 1), I will move on to the next task.

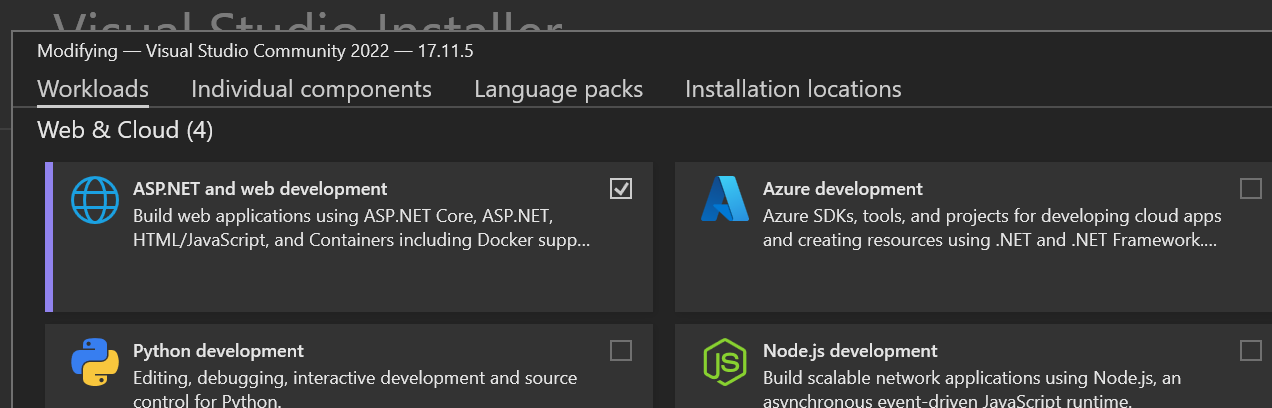


Рис. 1 – Required packages installed

# Завдання 2. Створення проектів

**Частина 1.**

1. Створіть проект для консольного додатку з назвою ConsoleToWeb з використанням (рис. 2) [dotnet CLI](https://learn.microsoft.com/en-us/dotnet/core/tools/dotnet-new)

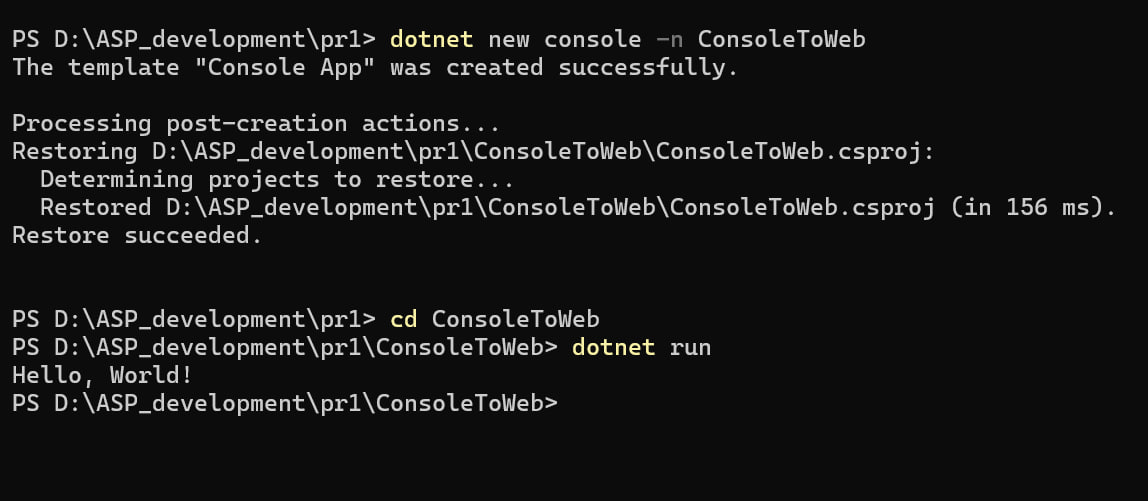


Рис. 2 – Create project

1. Перетворіть створений консольний додаток на веб-додаток (рис. 2)
2. Опишіть виконані кроки у звіті

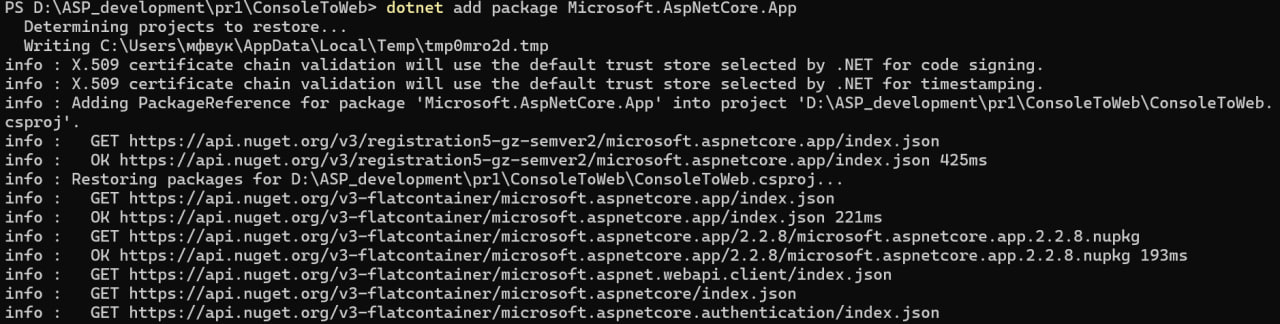


Рис. 2 – Convert to a web app

First, I added the necessary packages for the web application to function. Then I wrote the greeting itself (pic. 3)

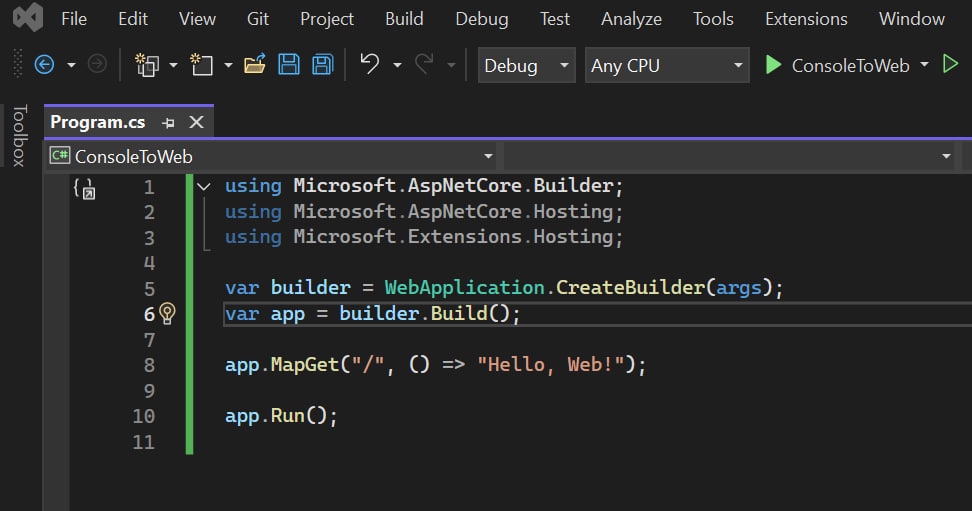


Рис. 3 – Привітання

Then from the console I ran the program (pic. 4)

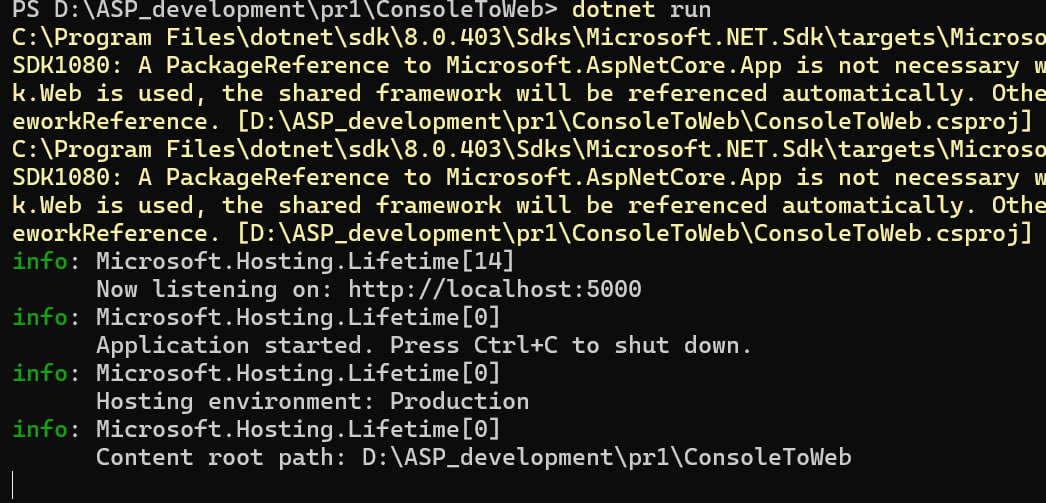


Рис. 4 – Launching a web application

And I followed the link: http://localhost:5000/ (pic. 5)

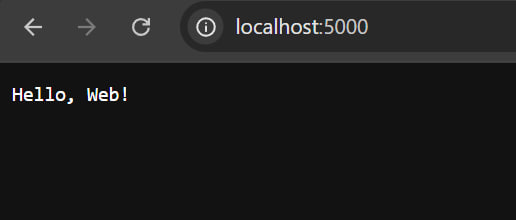


Рис. 5 – Перегляд веб-сторінки

**Частина 2.**

1. Створіть ASP.NET WebAPI проект без авторизації з назвою WebFromCli з використанням [dotnet CLI](https://learn.microsoft.com/en-us/dotnet/core/tools/dotnet-new) (рис. 6)

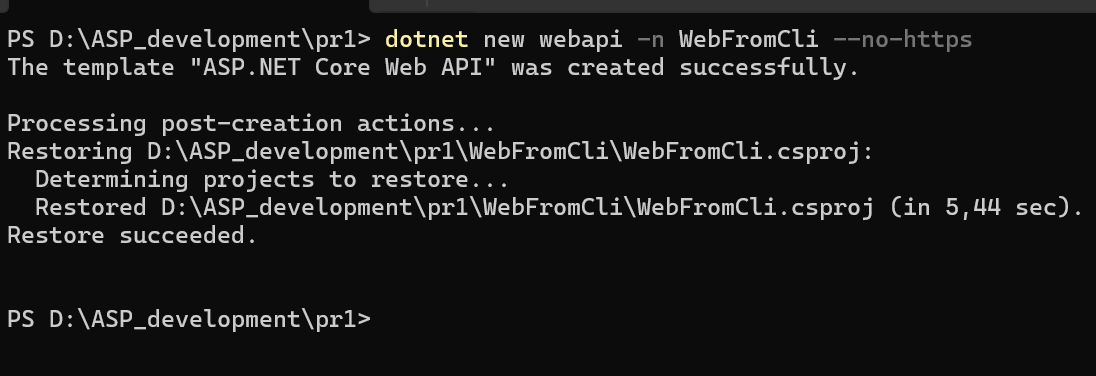


Рис. 6 – Create WebAPI project

1. Реалізуйте GET-ендпоінт “/who”, який повертатиме у відповідь ваше ім’я та прізвище (рис. 7)

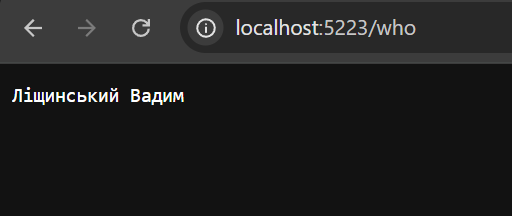


Рис. 7 – Implementing the "/who" endpoint

1. Реалізуйте GET-ендпоінт “/time”, який повертатиме у відповідь поточний час на сервері (рис. 8)

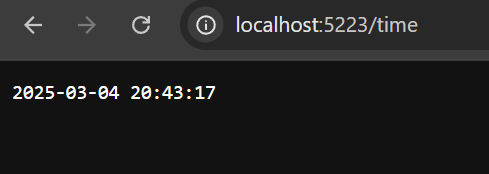


Рис. 8 – Implementing the "/time" endpoint

1. Наведіть лістинг реалізованих обробників у звіті

Listing of the “/who” endpoint:

app.MapGet("/who", () => "Ліщинський Вадим");

Listing of the ‘/time” endpoint:

app.MapGet("/time", () => DateTime.Now.ToString("yyyy-MM-dd HH:mm:ss"));

**Частина 3.**

1. Створіть ASP.NET MVC проект будь-яким зручним способом (рис. 9).

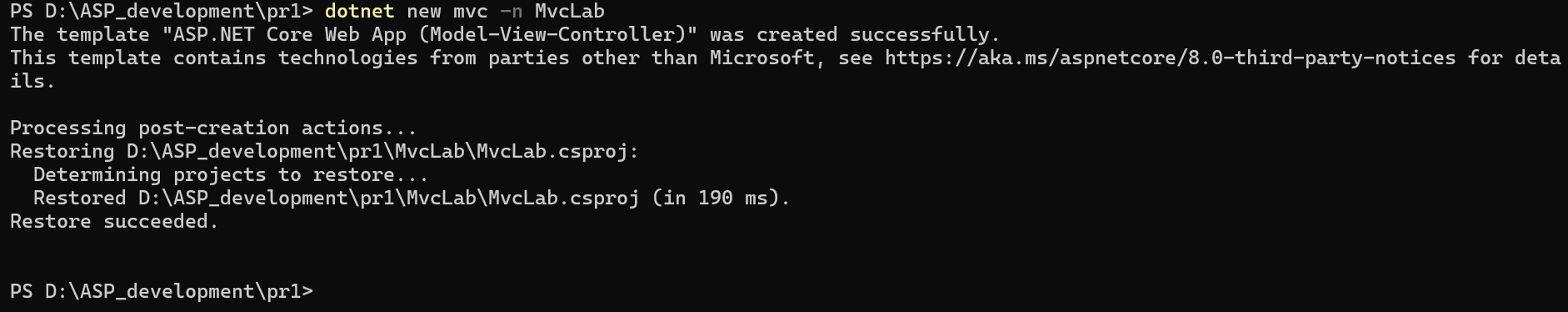


Рис. 9 – Create MVC project

1. Реалізуйте контролер з назвою LabController (рис. 10)

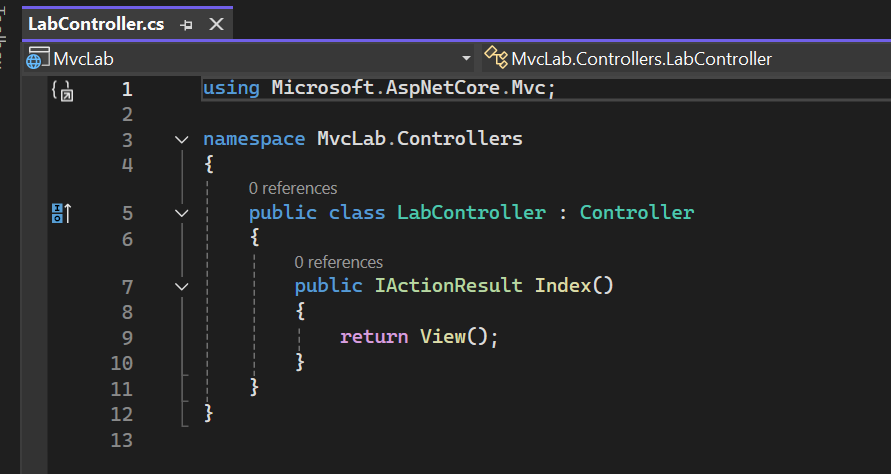


Рис. 10 – Controller implementation

1. В створеному контролері реалізуйте обробник /info, який повертатиме View з даними про номер лабораторної роботи, тему, мету та ім’я та прізвище виконавця в табличному вигляді (рис. 11)

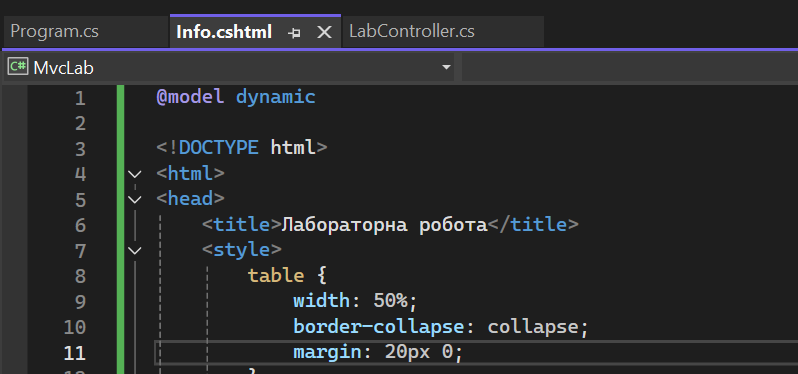


Рис. 11 – Handler /info

1. Дані для відображення передати з контролера (рис. 12)

I created a separate View (Info) to receive data from the LabController controller and set the default path in Program.cs to Lab/Info.cshtml (Lab/Info) so that when the web application is opened, the data is immediately received.

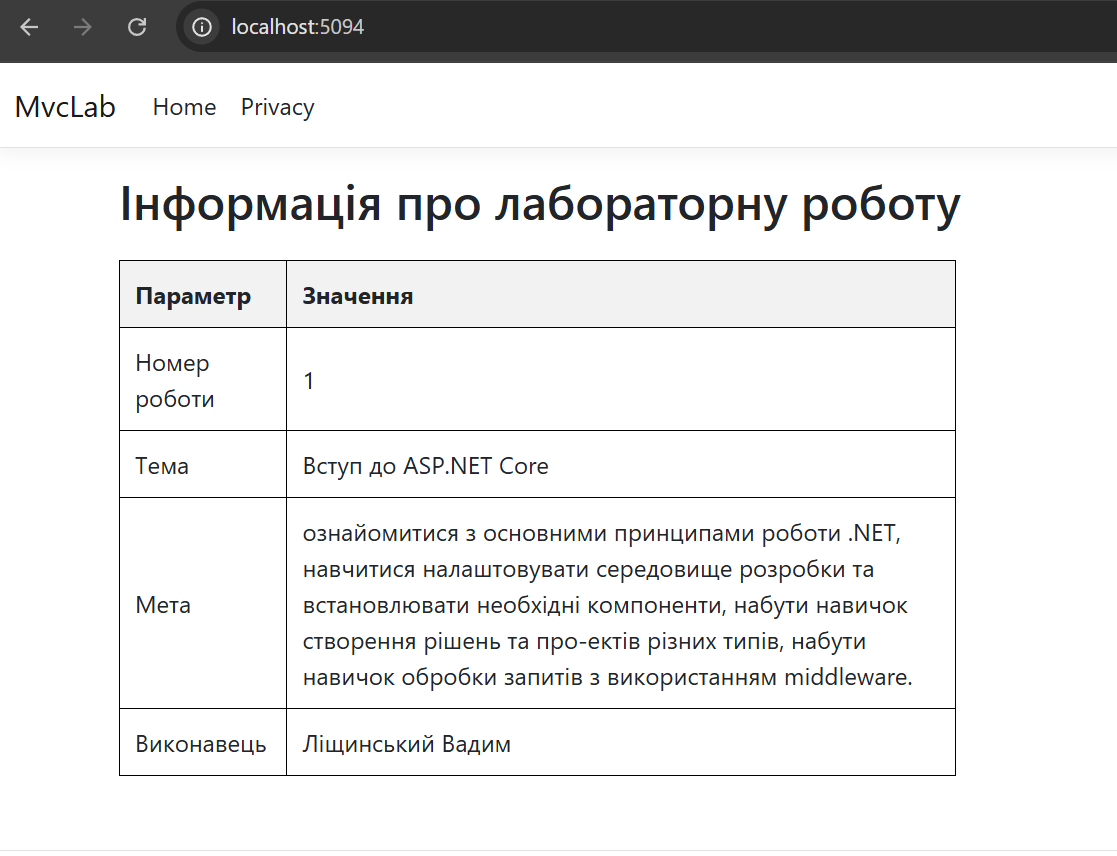


Рис. 12 – Information display

# Завдання 3. Робота з middleware

1. Ознайомтеся з [поняттям middleware](https://learn.microsoft.com/en-us/aspnet/core/fundamentals/middleware/?view=aspnetcore-9.0) в ASP.NET
2. Створіть ASP.NET WebAPI проект з назвою MiddlewareSandbox (рис. 13)

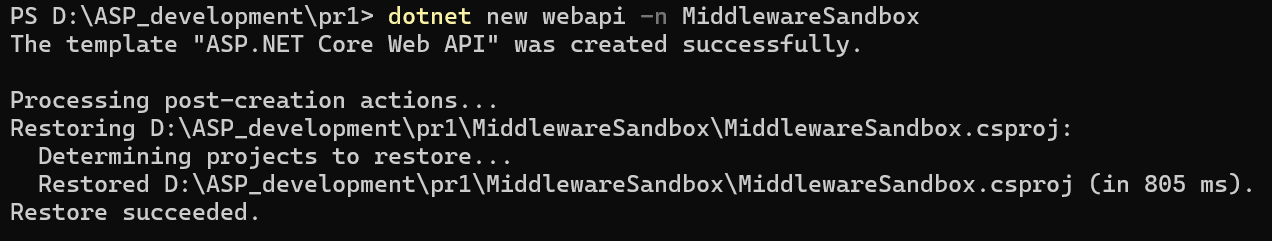


Рис. 13 – Create WebAPI project

1. Реалізуйте наступні middleware
   1. Реалізуйте middleware, яке рахує кількість запитів до сервера і повертає це число у відповіді, наприклад: The amount of processed requests is X.

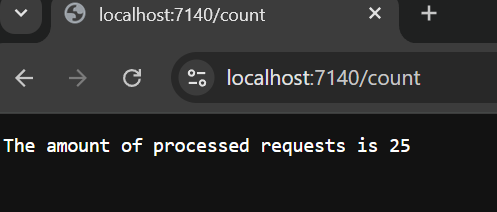


Рис. 14 –Counter implementation

* 1. Створіть middleware, яке аналізує параметри рядка запиту. Якщо в рядку запиту (query string) присутній параметр “custom”, то повертати у відповідь “You’ve hit a custom middleware!”, інакше пропускати запит далі.(рис. 15)

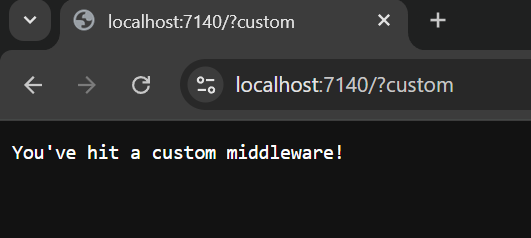


Рис. 15 –Customer implementation

* 1. Створіть middleware, яке логуватиме у консоль інформацію про всі запити. В логах відображати метод запиту (GET, POST тощо) та його шлях (/user, /product тощо). Для перевірки POST та інших методів запиту можна скористатися Postman або іншою подібною утилітою. (рис. 16)

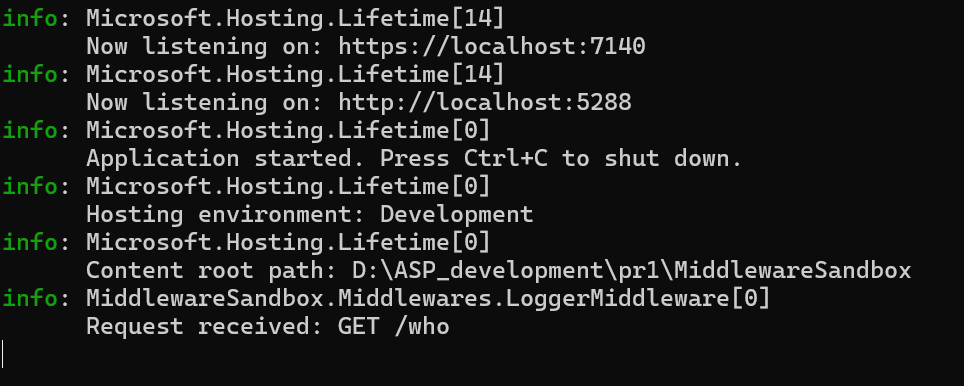


Рис. 16 –Logger implementation

* 1. Реалізуйте middleware, яке перевіряє наявність API-ключа у заголовках запиту. Для цього слід Перевіряти, чи є в запиті заголовок X-API-KEY. Якщо ключ неправильний або відсутній, повертати 403 Forbidden, не передавати запит для виконання далі. Якщо ключ в заголовку запиту співпадає з тим, який заданий на сервері, передавати запит далі. Для додавання відповідного запиту можна скористатися Postman або подібними утилітами. (рис. 17)

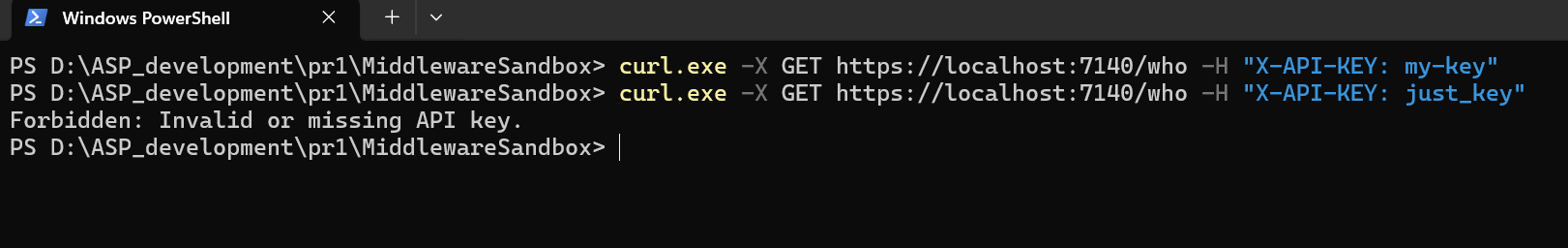


Рис. 17 –UseApi implementation

**Conclusions:** During the laboratory work, I got acquainted with the basic principles of .NET robots, learned how to configure the development environment and install the necessary components, acquired skills in creating solutions and projects of various types, and acquired skills in processing requests using middleware.