

# ЛАБОРАТОРНА РОБОТА № 1

## Початок роботи з ASP.NET Core 7. Middleware

**Мета роботи:** ознайомитися зі структурою ASP.NET Core 7 додатків, навчитися створювати найпростіші та використовувати наявні компоненти middleware у ASP.NET Core 7.

### Хід роботи

Завдання 1. Створення проекту ASP.Net Core Empty в VisualStudio та реалізація статичного сервера (створення папок wwwroot та static). Реалізація роботи сторінок по url: “/”, “/index.html” та “/files”.

Лістинг програми:

Program.cs:

```
var builder = WebApplication.CreateBuilder(args);
var app = builder.Build();

app.MapGet("/", () => "Hello, World!");

app.UseFileServer();
app.UseFileServer(new FileServerOptions
{
    EnableDirectoryBrowsing = true,
    FileProvider = new PhysicalFileProvider(
        Path.Combine(Directory.GetCurrentDirectory(), @"static"))
});

app.Run();
```

page.html:

```
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title></title>
</head>
<body>
  <h1>index.html</h1>
</body>
</html>
```

Результат виконання програми:

					ДУ «Житомирська політехніка».23.121.16.000 – Лр1			
Змн.	Арк.	№ докум.	Підпис	Дата				
Розроб.		Осипчук А.О.			Звіт з лабораторної роботи		Лім.	Арк.
Перевір.		Чижмотря О.В						1
Керівник								7
Н. контр.							ФІКТ Гр. ІПЗ-21-3[2]	
Зав. каф.								

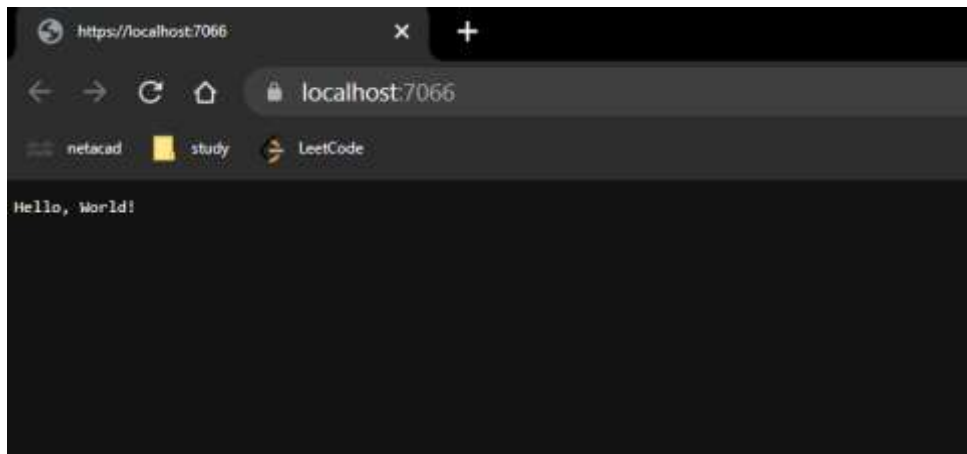


Рис. 1. https://localhost:7066/

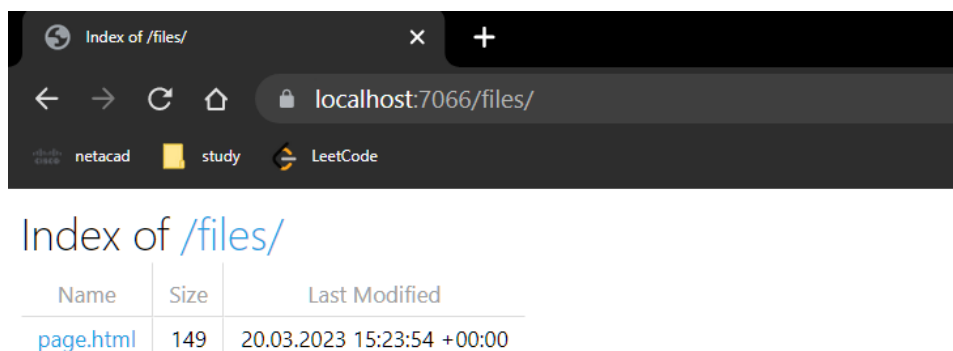


Рис. 2. https://localhost:7066/files

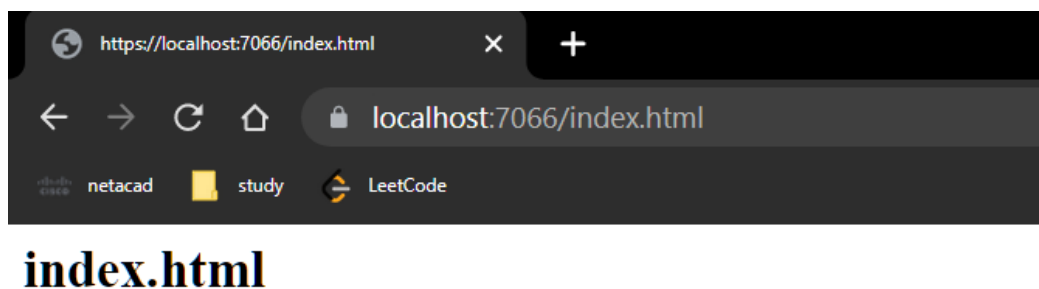


Рис. 3. https://localhost:7066/index.html

Завдання 2. Реалізуйте виведення повідомлення про виключення, які потенційно можуть статися у додатку.

Лістинг програми:

```
app.UseStatusCodePagesWithReExecute("/404.html");
```

Результат виконання програми:

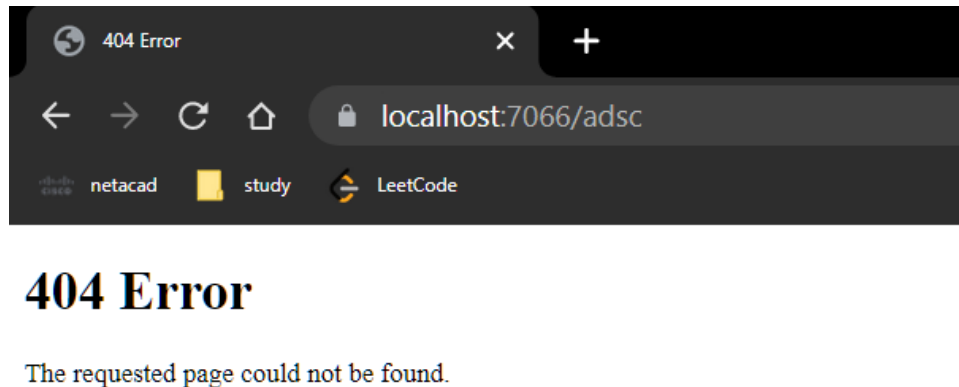


Рис. 4. Помилка 404

Завдання 3. За допомогою методів Map реалізуйте обробники маршрутів /home/index та /home/about.

Лістинг програми:

```
app.Map("/home", home =>
{
    home.Map("/index", Index);
    home.Map("/about", About);
});

void Index(IApplicationBuilder appBuilder)
{
    appBuilder.Run(async context => await context.Response.WriteAsync("Index"));
}
void About(IApplicationBuilder appBuilder)
{
    appBuilder.Run(async context => await context.Response.WriteAsync("About"));
}
```

Результат виконання програми:

		Осипчук А.О.			ДУ «Житомирська політехніка».23.121.16.000 – Лр1	Арк.
		Іужмотря О.В				3
Змн.	Арк.	№ докум.	Підпис	Дата		

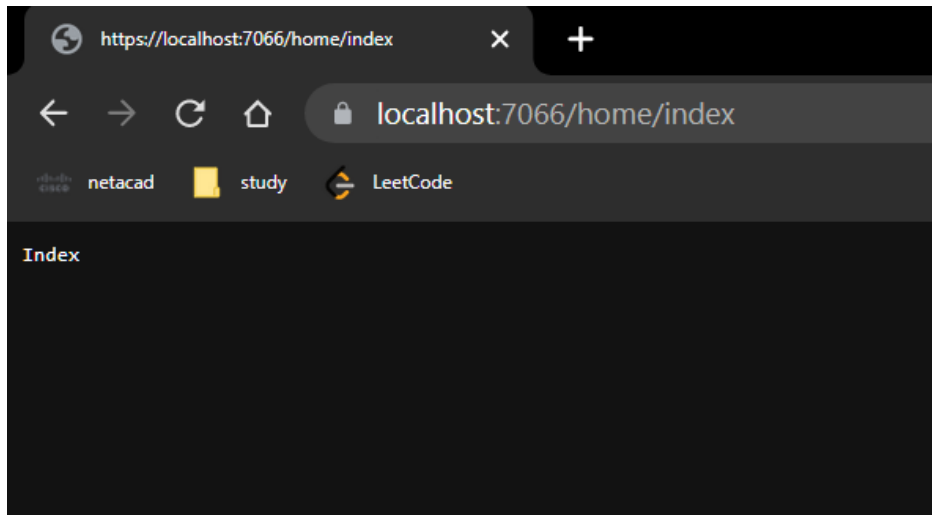


Рис. 5. /home/index

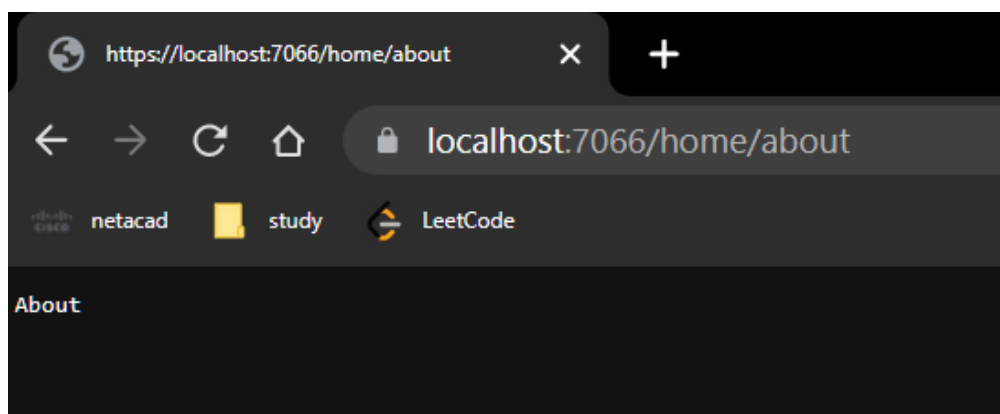


Рис. 6. /home/about

Завдання 4. Створіть та підключіть у проєкті власний компонент з назвою LoggerMiddleware, який буде записувати інформацію про усі запити до вашого сайту у текстовий файл з назвою access.txt.

Лістинг програми:

```
namespace AspDotNetLab1.middlewares
{
    public class LoggerMiddleware
    {
        private readonly RequestDelegate _next;

        public LoggerMiddleware(RequestDelegate next)
        {
            _next = next;
        }

        public async Task InvokeAsync(HttpContext context)
        {
            File.AppendAllText("access.txt", $"{DateTime.Now} {context.Request.Path}\n");

            await _next(context);
        }
    }
}
```

		Осипчук А.О.			ДУ «Житомирська політехніка».23.121.16.000 – Лр1	Арк.
		Лижмоторя О.В				4
Змн.	Арк.	№ докум.	Підпис	Дата		

}

Результат виконання програми:

```

tml    access.txt  X PageNotFound.cshtml  ErrorCont
1      23.03.2023 20:57:47 /
2      23.03.2023 20:57:51 /asdc
3      23.03.2023 20:58:27 /index.html
4      23.03.2023 20:58:46 /home/about
5      23.03.2023 20:58:50 /home/index
6      23.03.2023 20:58:53 /home/index
7      23.03.2023 20:58:59 /
8      23.03.2023 20:59:04 /home
9      23.03.2023 20:59:57 /
10     23.03.2023 21:00:27 /
11     23.03.2023 21:00:33 /secret-13412345
12     23.03.2023 21:01:55 /secret-1234
13     23.03.2023 21:02:16 /secret-1234
14     23.03.2023 21:02:20 /secret-1234
15     23.03.2023 21:02:20 /secret-1234
16     23.03.2023 21:02:36 /
17     23.03.2023 21:02:48 /secret
18     23.03.2023 21:02:54 /secret-1234
19     23.03.2023 21:03:26 /secret-1234
20     23.03.2023 21:03:34 /
21     23.03.2023 21:03:42 /secret-12345
22     23.03.2023 21:03:46 /secret12345
23     23.03.2023 21:03:49 /secre
24     23.03.2023 21:03:56 /asdcsecret
25     23.03.2023 21:04:48 /
26     23.03.2023 21:04:52 /secas
27     23.03.2023 21:04:59 /secret
28     23.03.2023 21:07:36 /
29     23.03.2023 21:07:47 /
30     23.03.2023 21:08:38 /
31     23.03.2023 21:08:42 /secret
32     23.03.2023 21:08:46 /secretasdcmv
33     23.03.2023 21:12:14 /
34     23.03.2023 21:12:18 /secret
35     23.03.2023 21:12:39 /
36     23.03.2023 21:12:42 /secret
37     23.03.2023 21:12:45 /secret-askj
38     23.03.2023 21:12:49 /secret1234
39     23.03.2023 21:13:58 /

```

Рис. 7. access.txt

Завдання 5. Створіть та підключіть у проєкті власний компонент з назвою SecretMiddleware, який буде відловлювати декілька маршрутів, наприклад “/secret-571743235872348” і виводити певне повідомлення.

Лістинг програми:

```

using System.Text.RegularExpressions;

namespace AspDotNetLab1.middlewares
{
    public class SecretMiddleware
    {
        private readonly RequestDelegate _next;

        public SecretMiddleware(RequestDelegate next)
        {
            _next = next;
        }

        public async Task InvokeAsync(HttpContext context)
        {

```

		Оситчук А.О.			ДУ «Житомирська політехніка».23.121.16.000 – Лр1	Арк.
		Іижмотря О.В				5
Змн.	Арк.	№ докум.	Підпис	Дата		

```

Regex validateSecretRegex = new Regex("^/secret");
Console.WriteLine();

if (validateSecretRegex.IsMatch(context.Request.Path))
{
    await context.Response.WriteAsync("This is a secret message.");
}
else
{
    await _next(context);
}
}
}
}

```

Результат виконання програми:

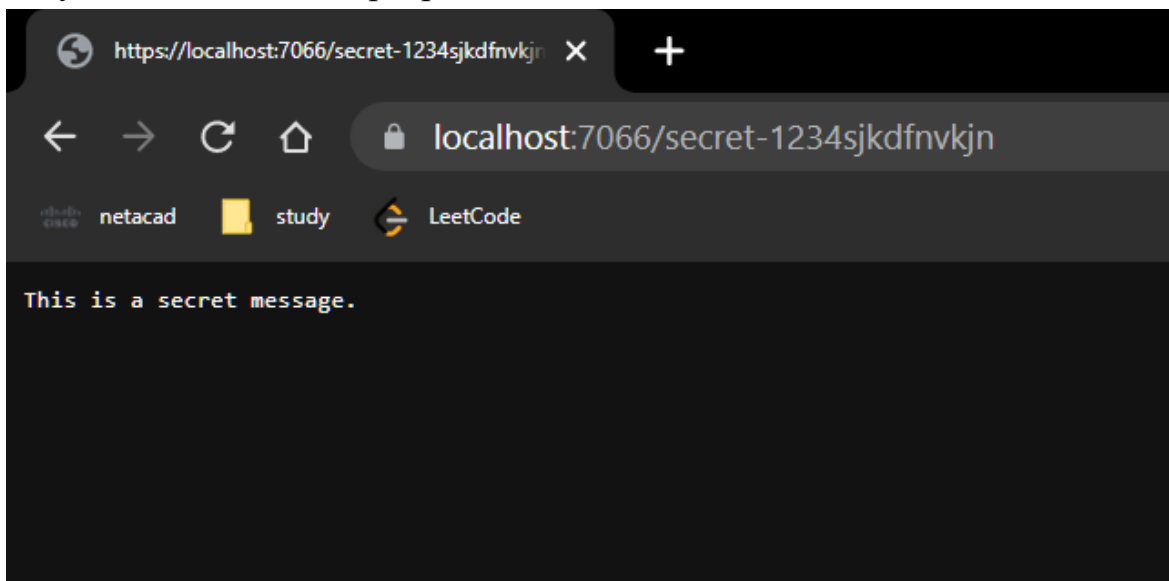


Рис. 8. /secret-

**Висновки:** на цій лабораторній роботі я ознайомився зі структурою ASP.NET Core 7 додатків, навчився створювати найпростіші та використовувати наявні компоненти middleware у ASP.NET Core 7.

		Осипчук А.О.			ДУ «Житомирська політехніка».23.121.16.000 – Лр1	Арк.
		Лижмоторя О.В.				6
Змн.	Арк.	№ докум.	Підпис	Дата		

## Додатки

### Program.cs

```
using AspDotNetLab1.middlewares;  
using Microsoft.AspNetCore.Builder;  
using Microsoft.Extensions.FileProviders;  
  
var builder = WebApplication.CreateBuilder(args);  
var app = builder.Build();  
  
app.UseStatusCodePagesWithReExecute("/404.html");  
  
app.MapGet("/", () => "Hello, World!");  
  
app.Map("/home", home =>  
{  
    home.Map("/index", Index);  
    home.Map("/about", About);  
});  
  
app.UseFileServer();  
app.UseFileServer(new FileServerOptions  
{  
    EnableDirectoryBrowsing = true,  
    FileProvider = new PhysicalFileProvider(  
        Path.Combine(Directory.GetCurrentDirectory(), @"static"))  
});  
app.UseMiddleware<SecretMiddleware>();  
app.UseMiddleware<LoggerMiddleware>();  
  
app.Run();  
  
void Index(IApplicationBuilder appBuilder)  
{  
    appBuilder.Run(async context => await context.Response.WriteAsync("Index"));  
}  
void About(IApplicationBuilder appBuilder)  
{  
    appBuilder.Run(async context => await context.Response.WriteAsync("About"));  
}
```

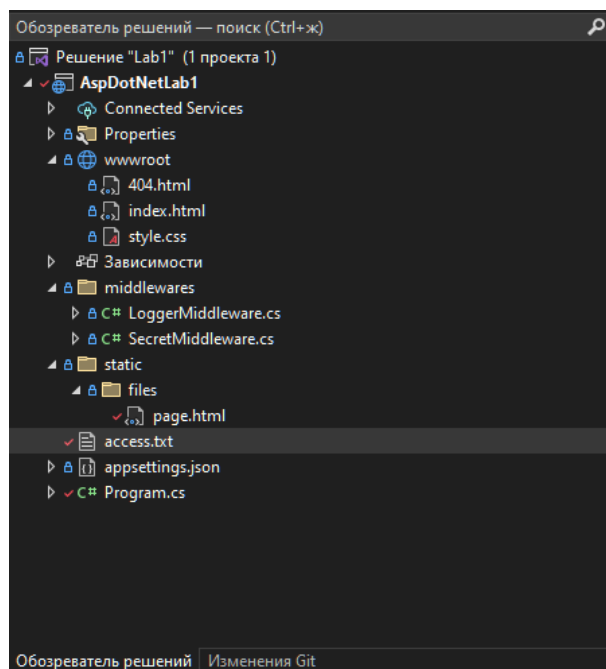


Рис. 9. Кінцева директорія додатку

		Осипчук А.О.			ДУ «Житомирська політехніка».23.121.16.000 – Лр1	Арк.
		Лижмотря О.В.				7
Змн.	Арк.	№ докум.	Підпис	Дата		