

Лабораторная работа № 10

Логирование, Middleware.

1. Цель работы.

Знакомство с библиотекой Serilog. Изучение работы компонентов Middleware.

2. Общие сведения.

Информацию об использовании Serilog см. здесь:
<https://github.com/serilog>

3. Выполнение работы

3.1. Исходные данные

Используйте проект из лабораторной работы №9.

3.2. Подготовка проекта

Добавьте в основной проект пакет NuGet «Serilog.AspNetCore»

Настройте конфигурацию Serilog в классе Program (см. <https://github.com/serilog/serilog-settings-configuration>).

Обязательно: настройки Serilog должны храниться в файле appsettings.json

3.3. Задание

Требуется выводить в консоль и сохранять в файле информацию обо **ВСЕХ** запросах, ответ на которые имеет код состояния отличный от 2XX (Ok).

Информация должна содержать запрошенный Url и код состояния:

```
2023-07-03 11:19:49.996 +03:00 [INF] ---> request /menu returns 404
2023-07-03 11:23:00.364 +03:00 [INF] ---> request /Cart/add/1 returns 302
```

3.3.1. Рекомендации к заданию

Чтобы не выполнять логирование во всех контроллерах и страницах основного проекта, опишите компонент Middleware, который будет выполнять проверку ответа и при необходимости логирование.

4. Вопросы для самопроверки

1. Для чего используются Moddleware?
2. Для чего используется объект `RequestDelegate`, передаваемый в конструктор класса `Middleware`.
3. Как работает конвейер обработки запросов `Http`?
4. Может ли `Middleware` обработать ответ, передаваемый клиенту?
5. Как добавить `Middleware` в конвейер обработки запросов `Http`?