ПИС-3, ПОИТ

**Лабораторная работа 2**

**ASPA** – приложение ASP.NET CORE

♣ – задания, требующие самостоятельного изучения (нет в лекциях)

**Задание 1.**

1. Разработайте ASPA, применив следующий шаблон.

Имя решения: **ASPA**

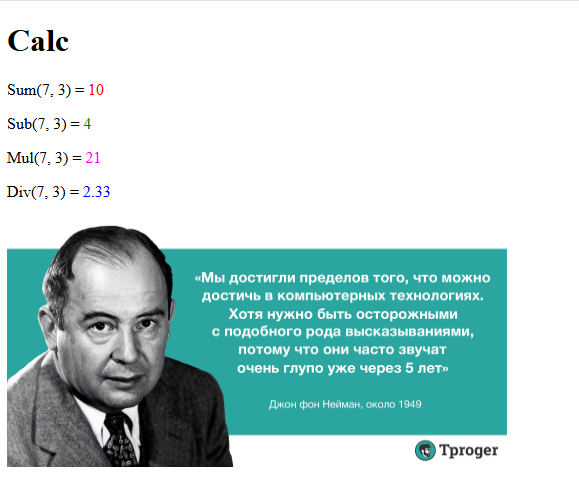
Имя проекта: **ASPA002\_1**

Версия .NET: **7.0** или **8.0**



1. Подключите к конечной точке **/aspnetcore** встроенный Middleware-элемент, отображающий приветственную страницу ASP.NET Core. Поясните Middleware-механизм на этом примере. Поясните: почему следующий ниже по коду обработчик **MapGet(“/aspnetcore”,…)** будет не работоспособным.
2. Подключите к конечной точке “**/”** встроенный Middleware-элемент, обрабатывающий статические файлы.
3. Разработайте файл **Index.html**-файл (рисунки ниже, файл ***Neumann.jpg*** прилагается).





1. Сделайте **Index.html** стартовой страницей.
2. С помощью браузера сделайте запрос к **ASPA002,** приводящийк следующему результату.



**Задание 2.**

1. Разработайте ASPA, применив следующий шаблон.

Имя решения: **ASPA**

Имя проекта: **ASPA002\_2**

Версия .NET: **7.0** или **8.0**



1. Сделайте приложение **ASPA002\_2** аналогичным приложению **ASPA002\_1.**
2. ♣ Внесите следующие изменения в приложение **ASPA002\_2:**
3. переименуйте страницу ***Index.html*** в ***Neumann.html***и при этом сделайте её стартовой  **(**требуется изменить список имён стартовых страниц**);**
4. подключите к конечной точке “**/static”** встроенный Middleware-элемент, обрабатывающий статические файлы;
5. кроме стандартного (***wwwroot***) директория сделайте ещё один (дополнительный) директорий с именем ***Picture*** предназначенный для статических файлов; перенесите в него файл ***Neumann.jpg***;
6. внесите необходимые изменения в ***Neumann.html.***

**Задание 3.**

1. Разработайте ASPA, применив следующий шаблон.

Имя решения: **ASPA**

Имя проекта: **ASPA002\_3**

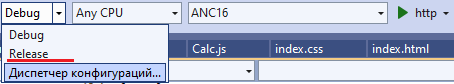
Версия .NET: **7.0** или **8.0**



1. ♣ Исследуйте следующий программный код web-сервера. Сервер имеет 5 конечных точек. Отправка запроса на 3 из них, приводит к исключению. Все исключения обрабатываются с помощью ***UseEcxeptionHandler***.



1. ♣ В рамках **ASPA002\_3** повторите код. Запустите приложение и помощью браузера отправьте запросы на конечные точки, вызывающие исключения. Исследуйте выводимый журнал, выводимый на консоль.
2. ♣ Объясните логику работы **ASPA002\_3.**
3. **Внимание!** Обработчик ошибок работает корректно в режиме Release.



**Задание 4.Ответьте на вопросы**

1. Поясните назначение паттерна Builder. Укажите: в каком месте ASPA применяется этот паттерн.
2. Поясните назначение паттерна Chain of Responsibility. Укажите: в каком месте ASPA применяется этот паттерн.
3. Поясните понятие: Middleware.
4. Поясните понятие: статический файл.
5. Какой отличительный (общепринятый) признак наименования, подключаемого Middleware-компонента.
6. Перечислите известные вам встроенные Middleware-компоненты.
7. Какое имя директория ASPA (по умолчанию), применяемого для хранения статических файлов приложения.
8. ♣ Кто такой Джон фон Нейман?☺

**Задание 4.Ответьте на вопросы**

1. Поясните назначение паттерна Builder. Укажите: в каком месте ASPA применяется этот паттерн.

**Назначение**: Паттерн "Строитель" используется для того, чтобы создать сложные объекты пошагово. Он позволяет конструировать объекты без указания их конкретных классов. Строитель разделяет процесс создания объекта и его представление, что позволяет создавать различные варианты объекта, используя один и тот же процесс.

**Применение в ASP.NET Core**: Паттерн Builder часто применяется в конфигурации приложений, где необходимо настроить различные параметры, например, через WebApplicationBuilder. Он позволяет пошагово настраивать параметры сервера, добавлять службы и настраивать middleware, не привязываясь к конкретной реализации.

1. Поясните назначение паттерна Chain of Responsibility. Укажите: в каком месте ASPA применяется этот паттерн.

**ASP.NET CORE: Middleware:** цепочка программных объектов, предназначенных для последовательной обработки запросов и ответов, реализация паттерна **Chain of Responsibility**.

1. Поясните понятие: Middleware.

**ASP.NET CORE: Middleware:** промежуточное программное обеспечение – компонент ASP.NET CORE–приложения**.**

**ASP.NET CORE: Middleware:** двунаправленный конвейер обработки запросов (прямое направление) и ответов (обратное направление).

Middleware в ASP.NET Core — это компоненты, которые обрабатывают HTTP-запросы. Они могут быть использованы для выполнения действий на запросах и ответах, таких как аутентификация, логирование, обработка ошибок, настройка CORS, маршрутизация и т. д. Каждый компонент в цепочке middleware выполняет свою задачу и передает запрос следующему компоненту.

1. Поясните понятие: статический файл.

Статический файл — это файл, который не изменяется на сервере и отдается клиенту "как есть". Обычно это изображения, CSS, JavaScript и другие ресурсы, которые приложение отдает напрямую клиенту без каких-либо изменений или логики обработки.

UseDefaultFiles, UseStaticFiles, UseWelcomePage, UseRouting, UseEndpoints, UseLogging, UseExceptionHandler

1. Какой отличительный (общепринятый) признак наименования, подключаемого Middleware-компонента.

**ASP.NET CORE: Middleware: *app*.Use*XXXXX*()**– имена Middleware-элнмент: **app.UseHttpLogging()**.

1. Перечислите известные вам встроенные Middleware-компоненты.
2. Какое имя директория ASPA (по умолчанию), применяемого для хранения статических файлов приложения. wwwroot
3. ♣ Кто такой Джон фон Нейман?☺

Джон фон Нейман (1903–1957) — выдающийся венгерско-американский математик, физик, инженер и компьютерный ученый. Он разработал архитектуру компьютеров, которая называется **архитектура фон Неймана**, и является основой для большинства современных компьютеров. Также он сделал значительный вклад в теорию игр, квантовую механику, а также участвовал в разработке атомной бомбы.