Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ

УНИВЕРСИТЕТ»

«Программирование инетренет-серверов»

Отчёт по лабораторной работе №1

ASPA.

Выполнил: Сосновец М.И.

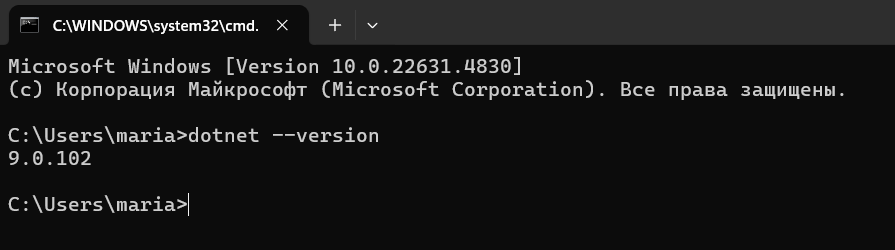
ФИТ 3 курс 4 группа

Преподаватель: Гончар Е.?.

Минск 2025

**Задание 1.**

1. Определите версию .NET, установленного на ваш компьютер.



**Задание 2.**

1. Разработайте простейшее ASPA, применив шаблон следующий шаблон.

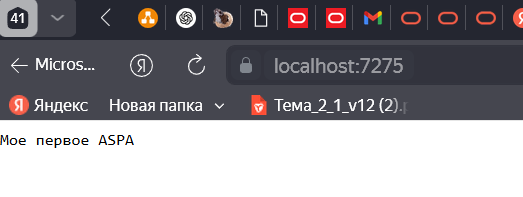
Имя решения: **ASPA**

Имя проекта: **ASPA001**

Версия .NET: **7.0** или **8.0**

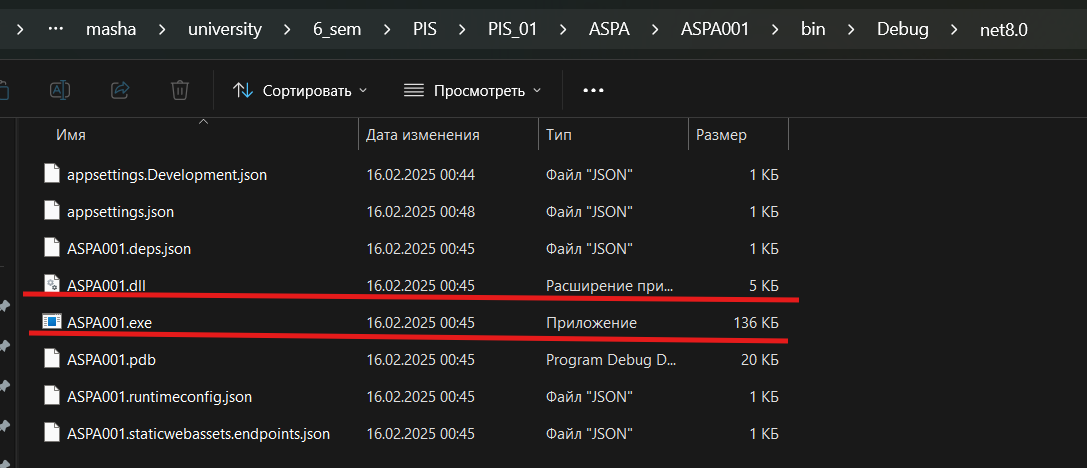
1. Приложение должно выводить в окно браузера следующее сообщение.

|  |
| --- |
| var builder = WebApplication.CreateBuilder(args);  var app = builder.Build();  app.MapGet("/", () => "Мое первое ASPA");  app.Run(); |

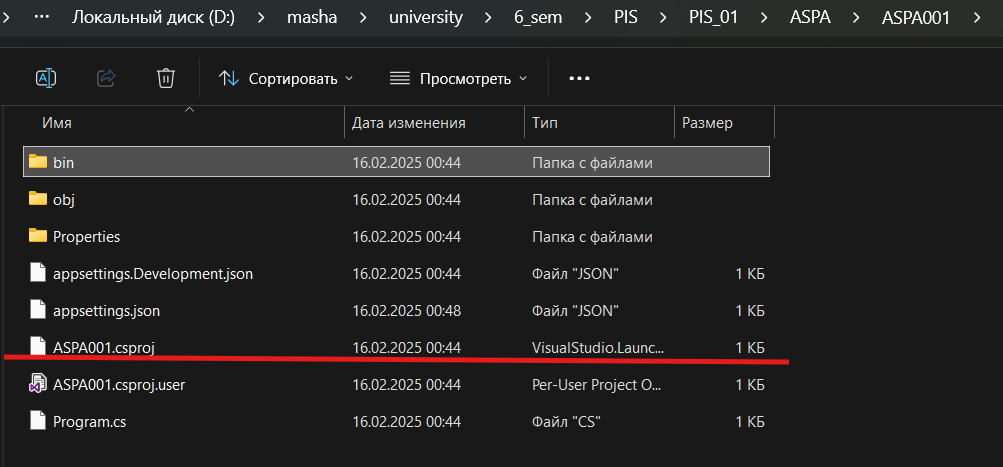


**Задание 3.**

1. Найдите в проекте **ASPA001** исполняемыефайлы **ASPA001.exe** и **ASPA001.dll.**



1. Найдите файл **ASPA001.cproj** и поясните его назначение.



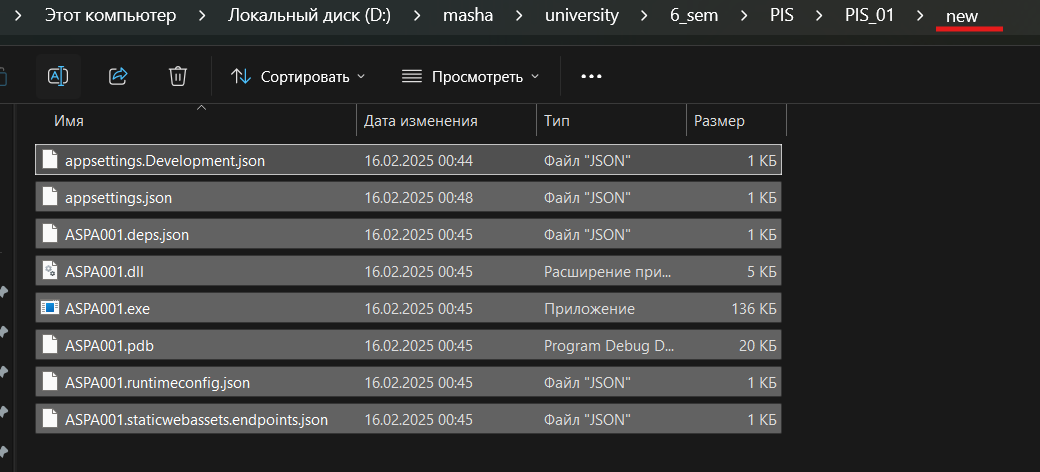
файл проекта, который содержит информацию о зависимостях, целевой версии .NET, конфигурации сборки и других параметрах проекта.

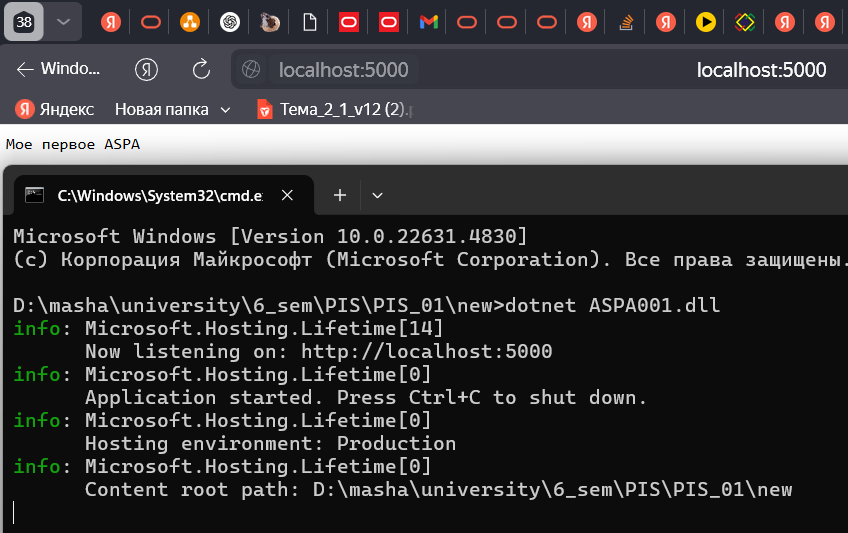
1. Пересоберите приложение **ASPA001** в режиме **Release**.
2. Объясните разницу между режимам**и Release и Debug.**

**Debug —** режим отладки, включает дополнительную информацию о отладке, что помогает разработчику в процессе тестирования.

**Release —** оптимизированный режим, предназначен для развертывания, файлы меньше по размеру и работают быстрее.

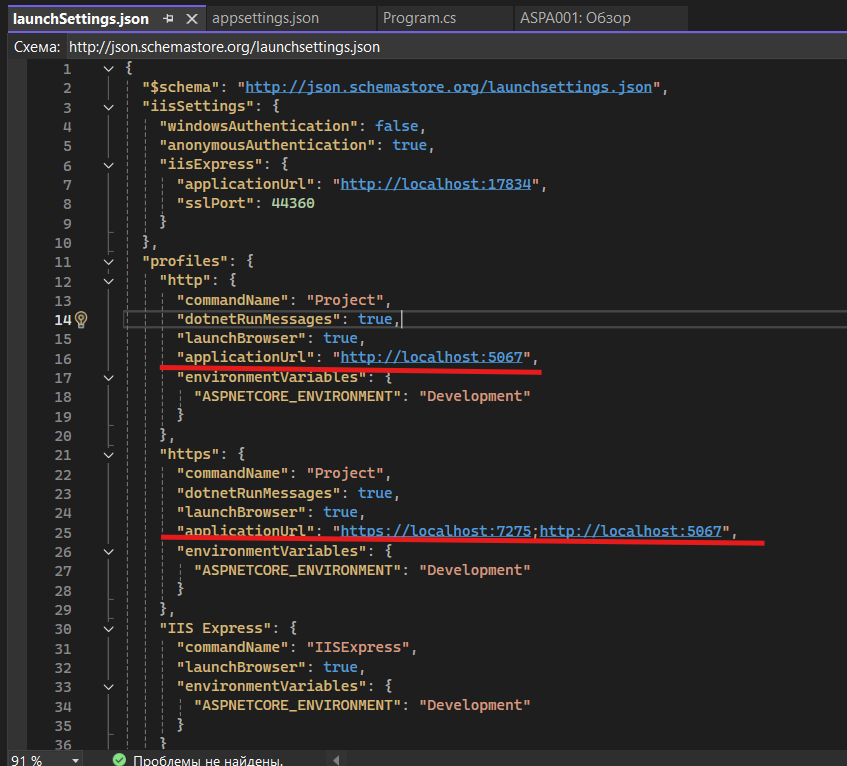
1. Перенесите из проекта ASPA001вотдельную папку только необходимые файлы и запустите приложение без помощи VStudio (с помощью файла ASPA001.exe, с помощью утилиты dotnet).

****

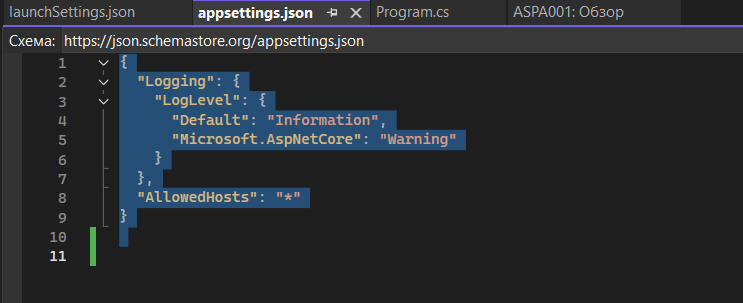
****

2. Найдите конфигурационный файл **launchSettings.json** поясните его назначение. Укажите, место в котором задаётся TCP-порт.

Файл launchSettings.json используется в приложениях ASP.NET Core для настройки параметров запуска и отладки приложения. Он определяет различные профили, которые могут использоваться при запуске проекта, а также настройки для IIS и IIS Express.



1. Найдите конфигурационный файл **appsettings.json** поясните его назначение.



appsettings.json используется для хранения конфигурационных данных приложения, таких как строки подключения, параметры настройки и другие параметры.

**1. Logging**

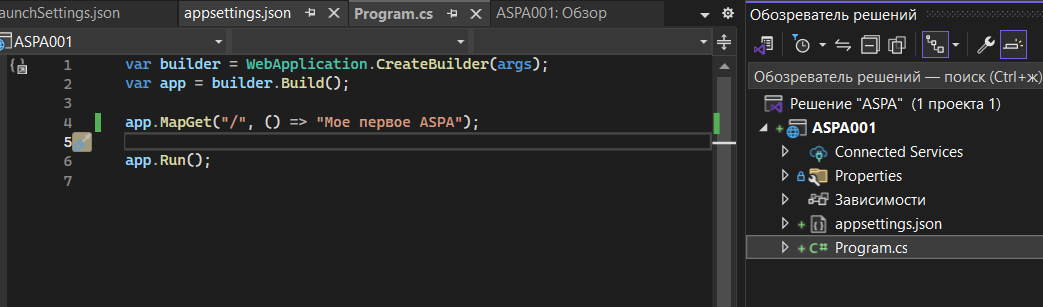
Этот раздел отвечает за настройки логирования в приложении.

* **LogLevel**: Определяет уровень логирования для различных категорий.
  + **Default**: Устанавливает уровень логирования по умолчанию. В данном случае задано значение "Information", что означает, что будут записываться все сообщения уровня информации и выше (например, предупреждения и ошибки).
  + **Microsoft.AspNetCore**: Устанавливает более высокий уровень логирования для категорий, относящихся к ASP.NET Core. Значение "Warning" означает, что будут записываться только сообщения уровня предупреждения и выше для компонентов ASP.NET Core (например, предупреждения и ошибки), а информационные сообщения не будут записываться.

**2. AllowedHosts**

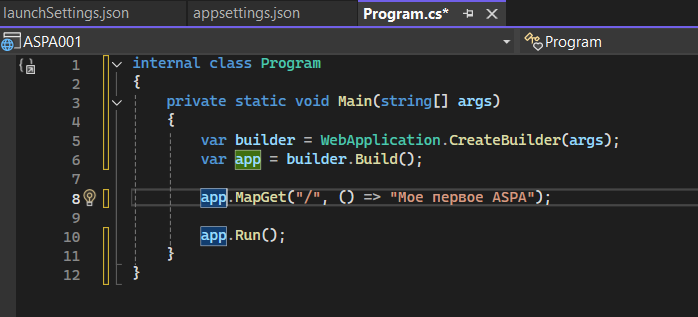
* **AllowedHosts**: Этот параметр определяет, какие хосты разрешены для вашего приложения. Значение "\*" означает, что разрешены все хосты, что может быть полезно для разработки, но не рекомендуется для производственной среды из соображений безопасности. В производстве лучше указывать конкретные разрешенные домены или IP-адреса.

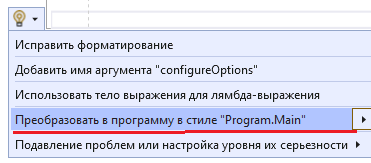
1. Найдите файл **Program.cs** поясните его назначение.



Program.cs — это основной файл приложения, который содержит метод Main, запускающий приложение и настраивающий его.

1. Преобразуйте код в стиль **Prorgam.Main**





1. Поясните комментариями каждую строку кода в **Program.cs**.

|  |
| --- |
| internal class Program  {  // Точка входа в приложение  private static void Main(string[] args)  {  // Создание и настройка веб-приложения на основе переданных аргументов  var builder = WebApplication.CreateBuilder(args);  // Построение приложения на основе настроек  var app = builder.Build();  // Определение маршрута для обработки GET-запроса на корневой URL ("/")  app.MapGet("/", () => "Мое первое ASPA");  // Запуск приложения  app.Run();  }  } |

**Задание 4.**

1. Подключите и настройте в ASPA001 сервис HTTPLogging
2. Убедитесь в его работоспособности.

