

# MVC. Контроллеры

**№ урока:** 8 **Курс:** ASP.NET Core Essential

**Средства обучения:** Visual Studio 2017

## Обзор, цель и назначение урока

Изучение понятия MVC-контроллера, его назначение. Создание контроллеров, определение маршрута контроллера, атрибуты. Передача файлов пользователям

## Изучив материал данного занятия, учащийся сможет:

- Понимать назначение контроллера
- Создавать свои контроллеры
- Использовать доступные атрибуты контроллеров
- Определять результаты действий контроллера
- Производить отправку на сторону клиента файлов с сервера

## Содержание урока

1. Понятие контроллера
2. Создание контроллера
3. Результаты действий контроллеров
4. Атрибуты
5. Ограничение типов запросов
6. Контекст контроллера
7. Отправка файлов с сервера

## Резюме

- **Контроллер** – специальный класс, содержащий в себе серию методов, ответственных за обработку запросов по определенному маршруту
- Методы контроллера принимают запрос, производят связанную с этим запросом определенную работу на сервере, и генерируют соответствующий **http-ответ**
- Все контроллеры должны находиться в папке **Controllers**
- Класс-контроллер должен обязательно наследоваться от системного класса **Controller** пространства имен **Microsoft.AspNetCore.Mvc**
- Название класса должно строиться по принципу **<name>Controller.cs**
- В следствии обработки поступившего запроса, контроллер производит и отправляет на сторону клиента определенный результат. Результаты бывают:
  - Объекты **ActionResult**, как правило – **html-представления**
  - **FileResult**, базовый класс для всех объектов, отправляющих файлы пользователю
  - Любой другой стандартный тип (**string**, **bool**, **int** и прочие)
- Атрибуты контроллеров задают дополнительные системные сведения о том или ином объекте контроллера, к которому данный атрибут был применен
- Атрибуты контроллеров задают дополнительные системные сведения о том или ином объекте контроллера, к которому данный атрибут был применен.
- Атрибуты бывают:
  - **[NonController]** – применяется по отношению к классу-контроллеру в случае, когда разработчик не желает, что бы система считала класс контроллером
  - **[NonAction]** – позволяет указать системе, что этот метод контроллера не применяется для обработки запросов
  - **[ActionName(string value)]** – применяется для переопределения имени метода-обработчика

- Так же, помимо указанных атрибутов, существуют особые, которые применяются для указания методу, какие конкретно типы **http-запросов** должен обрабатывать тот
- Всего различают основные:
  - **[HttpGet]**
  - **[HttpPut]**
  - **[HttpPost]**
  - **[HttpDelete]**
- Контекст контроллера – эта вся связанная с конкретным контроллером системная информация, разбитая на условные разделы
- Всего различают следующие разделы:
  - **HttpContext** – собственно говоря, сам контекст контроллера
  - **ActionDescriptor** – описание производимого действия
  - **ModelState** – валидация данных, отправленных пользователю
  - **RouteData** – информация о маршруте
- Отправить файл клиенту можно следующими способами:
  - **FileContentResult** – отправляет массив байт, считанный с файла
  - **FileStreamResult** – отправляет файл посредством потока байт
  - **VirtualFileResult** – отправка файла по виртуальному пути
  - **PhysicalFileResult** – для отправки файла используется настоящий путь, по которому этот файл хранится на сервере

### Закрепление материала

- Что такое контроллер? Каково его применение?
- Расскажите о правилах создания контроллера
- Какие бывают результаты действий контроллера?
- Опишите атрибуты, которые могут применяться к контроллерам и их методам
- Как можно отправить файл клиенту?

### Дополнительное задание

#### Задание

Создайте контроллер, который будет обрабатывать маршрут `File\DownloadFile`. Путь должен быть дефолтным при загрузке сайта. В ответ должна высвечиваться html-форма. Форма будет обладать следующими полями: имя, фамилия, название файла. При нажатии на клавишу «продолжить», в качестве ответа пользователь должен получить txt-файл с указанным в форме названием. Внутри файла должны быть записаны введенные в форму имя и фамилия

### Самостоятельная деятельность учащегося

#### Задание 1

Создайте набор контроллеров, обрабатывающие свои маршруты и возвращающие в качестве ответа связанные с ними html-представления

#### Задание 2

Разместите на сервере файл `example.txt` и попробуйте отправить его клиенту любым желаемым способом

#### Задание 3

Зайдите на сайт MSDN.

Используя поисковые механизмы MSDN, найдите самостоятельно описание темы по каждому примеру, который был рассмотрен на уроке, так, как это представлено ниже, в разделе «Рекомендуемые ресурсы», описания данного урока. Сохраните ссылки и дайте им короткое описание.

### Рекомендуемые ресурсы

Официальная документация ASP.NET Core  
<https://docs.microsoft.com/en-us/aspnet/core/>

Контроллеры  
<https://metanit.com/sharp/aspnet5/5.1.php>