

Отправка файлов в ASP.NET MVC 5

Последнее обновление: 31.10.2015



Для отправки клиенту файлов предназначен **FileResult**. Однако так как это абстрактный класс, то фактически мы будем иметь дело с его наследниками:

- **FileContentResult**: отправляет клиенту массив байтов, считанный из файла
- **FilePathResult**: представляет простую отправку файла напрямую с сервера
- **FileStreamResult**: данный класс создает поток - объект `System.IO.Stream`, с помощью которого считывает и отправляет файл клиенту

Во всех трех случаях для отправки файлов применяется метод **File**, который и возвращает объект `FileResult`. Только в зависимости от выбранного способа используется соответствующая перегруженная версия этого метода.

Чтобы отправить файл из файловой системы (то есть использование объекта `FilePathResult`), нам надо указать в методе `File` три параметра: путь к файлу на стороне сервера, тип содержимого и имя файла для принимающей стороны (имя файла необязательно, и можно обойтись в принципе только двумя параметрами).

```
public FileResult GetFile()
{
    // Путь к файлу
    string file_path = Server.MapPath("~/Files/PDFIcon.pdf");
    // Тип файла - content-type
    string file_type="application/pdf";
    // Имя файла - необязательно
    string file_name = "PDFIcon.pdf";
    return File(file_path,file_type,file_name);
}
```

Предполагается, что у нас в проекте есть папка `Files`, в которой лежит файл `PDFIcon.pdf`. Метод **`Server.MapPath`** позволяет построить полный путь к ресурсу из каталога в проекте. Но также можно использовать и абсолютные пути, обращаясь к любому файлу в файловой системе, например, `string file_path = @"C:\Book\PDFIcon.pdf";`

И, при обращении, например, по пути `Home/GetFile` нам будет предложено сохранить данный файл на локальном компьютере.

Похожим образом работает и классы `FileContentResult`, только вместо имени файла в методе `File` указывается массив байтов, в который был считан файл:

```
// Отправка массива байтов
public FileResult GetBytes()
{
```

```

string path = Server.MapPath("~/Files/PDFIcon.pdf");
byte[] mas = System.IO.File.ReadAllBytes(path);
string file_type = "application/pdf";
string file_name = "PDFIcon.pdf";
return File(mas, file_type, file_name);
}

```



И если мы хотим вернуть объект `FileStreamResult`, то в качестве первого аргумента в методе `File` идет объект `Stream` для отправляемого файла:

```

// Отправка потока
public FileResult GetStream()
{
    string path = Server.MapPath("~/Files/PDFIcon.pdf");
    // Объект Stream
    FileStream fs = new FileStream(path, FileMode.Open);
    string file_type = "application/pdf";
    string file_name = "PDFIcon.pdf";
    return File(fs, file_type, file_name);
}

```

Дополнительные материалы

 Модуль 13. Отправка файлов в ASP.NET MVC 5 

[Назад](#) [Содержание](#) [Вперед](#)



29 Комментариев metanit.com

 Войти ▾

 Рекомендовать  Поделиться

Лучшее в начале ▾



Присоединиться к обсуждению...

ВОЙТИ С ПОМОЩЬЮ

ИЛИ ЧЕРЕЗ DISQUS 



Имя



Vladyslav Hrehul • 2 года назад

Какой смысл перекодировать файл в массив байтов? Безопасность? Объясните плиз.

1 ^ | ▾ • Ответить • Поделиться ›



Metanit Модератор ➔ Vladyslav Hrehul • 2 года назад

при отправке можно произвести некоторую постобработку файла

2 ^ | ▾ • Ответить • Поделиться ›



Саша Безгин • 6 месяцев назад



А как тестировать FileResult?

^ | v • Ответить • Поделиться ›



Алишер Есенкожа • год назад

как вернуть два файла без ZIP через FileResult

^ | v • Ответить • Поделиться ›



Metanit Модератор → Алишер Есенкожа • год назад

одномоментно можно передать только один файл

1 ^ | v • Ответить • Поделиться ›



Гость • год назад

Добрый день.

Подскажите, пожалуйста, а как правильно формировать путь к скачиваемому файлу динамически: например, если название скачиваемого файла приходит в метод в виде гет-параметра.

Если вот так написать:

```
public FileResult GetFile(string fileName)
{
```

```
    string file_path = Server.MapPath("~/Files/" + fileName);
```

```
    return File(file_path, System.Net.Mime.MediaTypeNames.Application.Octet);
}
```

то почему-то имя файла к пути не присоединяется. В отладчике file_path выглядит так "C:\\Documents and Settings\\Admin\\Рабочий стол\\MvcApplication1\\MvcApplication1\\Files\\"

Спасибо

^ | v • Ответить • Поделиться ›



Metanit Модератор → Гость • год назад

наверное параметр fileName пустой, посмотрите через отладку

^ | v • Ответить • Поделиться ›



Andrew • 2 года назад

Пробую обратиться к методу таким способом:

```
@Ajax.ActionLink("Load File", "GetFile", null)
```

Метод GetFile начинает выполняться, но отправка файла не происходит. Не пойму, почему?

^ | v • Ответить • Поделиться ›



Metanit Модератор → Andrew • 2 года назад

если не ошибаюсь, через ajax так не загрузить файл

^ | v • Ответить • Поделиться ›



Никита Савкин • 3 года назад

Здравствуйте.

А как лучше поступить, если на серверной стороне на основе данных не бл формируется

А как лучше поступить, если на серверной стороне на основе данных из од формируется Excel файл и для формирования требуется порядка 30 секунд? Подскажите в каком направлении смотреть хотя бы?

^ | v • Ответить • Поделиться ›



Metanit Модератор → Никита Савкин • 3 года назад

ну надо создать `xlsx`-файл и также его можно отправить как и любой другой файл

^ | v • Ответить • Поделиться ›



Никита Савкин → Metanit • 3 года назад

Просто пытался передавать его как двоичный файл, он почему то не после скачивания, а сохранять на сервере лишний раз, мне кажется не лучшей идеей. И еще метод наверно лучше асинхронным сделать?

^ | v • Ответить • Поделиться ›



Metanit Модератор → Никита Савкин • 3 года назад

для передачи можно использовать заголовок `application/octet-stream` если многие пользователи должны скачивать одну и ту же информацию, то файл лучше сохранять, чтобы потом не прибегать к средствам excel для нового формирования файла.

Да. метод лучше сделать асинхронным

2 ^ | v • Ответить • Поделиться ›



Дени → Metanit • год назад

Здравствуйте, у меня схожая ситуация только, каждый файл формируется для отдельного пользователя, т.е. сохранять его не хочу на сервере, когда то давно пробовал через stream но если вес файла превышал 150-200 мб выдавал ошибку, можно ли обойти это ограничение и чем это грозит? сервер под рукой.

^ | v • Ответить • Поделиться ›



Andrew → Никита Савкин • 2 года назад

Получилось ли передать файл, без сохранения на сервер? Поделитесь, пожалуйста, как сделали. Тоже над этим мучаюсь(

^ | v • Ответить • Поделиться ›



Никита Савкин → Andrew • 2 года назад

Извини, что долго.

Нет сделал сохранение, по причине указанной Metanit. Файл не каждый раз обязательно создавать.

^ | v • Ответить • Поделиться ›



Black Bull • 3 года назад

Здравствуйте.

А как поступить с реализацией отправкой файла размером более 500 Мб? при выполнении `GetFile()` ничего не происходит...

^ | v • Ответить • Поделиться ›



Дима • 3 года назад

А как прописать путь, если файл хранится в базе данных?

^ | v • Ответить • Поделиться ›



Metanit Модератор → Дима • 3 года назад

тогда вам путь не нужен, просто считываете из базы данных все байты файла в массив и вторым или третьим способом отправляете его пользователю

^ | v • Ответить • Поделиться ›



Дима → Metanit • 3 года назад

Подскажите, как считать из базы данных, какой метод использовать?

^ | v • Ответить • Поделиться ›



Metanit Модератор → Дима • 3 года назад

в следующей статье описано сохранение и получение картинок из бд - <http://metanit.com/sharp/ar...>

^ | v • Ответить • Поделиться ›



Igor Knysh • 3 года назад

Здравствуйте. А что делать в случае ошибки? По идее весь код должен быть заключен в try... catch. И что делаем в блоке catch?

^ | v • Ответить • Поделиться ›



Metanit Модератор → Igor Knysh • 3 года назад

в случае ошибки обычно логгирование делают, или иным образом обрабатывают ошибку, например, направляют пользователя на страницу с ошибкой

^ | v • Ответить • Поделиться ›



Igor Knysh → Metanit • 3 года назад

Решение найдено. Так как `FileResult` наследуется от `ActionResult`, делаем метод, возвращающий `ActionResult`. Остальное - детали. Спасибо.

^ | v • Ответить • Поделиться ›



Igor Knysh → Metanit • 3 года назад

т.е. получается обработка не в методе контроллера? Если в блоке `catch` нет `return` объекта соответствующего типа - компилятор не пропустит. Таким образом будет срабатывать некий глобальный обработчик. У меня случай, когда по введенным пользователем данным формируется и возвращается некий pdf-документ. Было-бы правильно в случае ошибок просто вернуть юзера к форме ввода.

^ | v • Ответить • Поделиться ›



Metanit Модератор → Igor Knysh • 3 года назад

вообще подобные ошибочные ситуации лучше проверять в коде с помощью условных конструкций, а не через `try..catch`. Например, валидация модели `if(ModelState.IsValid)` и если проверка не проходит, то отправляемся на ту же форму с выводом ошибки или куда-то еще перенаправляем. Также можно определить фильтр исключений,

который бы обрабатывал ошибку

^ | v • Ответить • Поделиться ›



Igor Knysh → Metanit • 3 года назад

Да, это понятно, спасибо.

^ | v • Ответить • Поделиться ›



W2 • 4 года назад

а как передать файлы с клиента на сервер?

^ | v • Ответить • Поделиться ›



Metanit Модератор → W2 • 4 года назад

<http://metanit.com/sharp/ar...>

^ | v • Ответить • Поделиться ›

ТАКЖЕ НА METANIT.COM

[Django | Определение форм](#)

[JavaScript | Стрелочные функции](#)

[Вконтакте](#) | [Twitter](#) | [Google+](#) | [Канал сайта на youtube](#) | [Помощь сайту](#)

Контакты для связи: metanit22@mail.ru

Copyright © metanit.com, 2012-2018. Все права защищены.