

آشنایی با مفهوم مسیریابی در Web API

در این قسمت با نحوه‌ی تناظر آدرس‌ها توسط Web API به متدهای موجود در Controller آشنا می‌شوید. در هر درخواستی که ارسال می‌شود، Web API، انتخاب مناسب را با رجوع به جدولی با نام جدول مسیرها انجام می‌دهد. زمانی که یک پروژه‌ی جدید با استفاده از ASP.NET MVC 4 ایجاد می‌کنید، یک route پیش فرض به صورت ذیل در متد RegisterRoutes قرار می‌گیرد.

```
routes.MapHttpRoute(
    name: "DefaultApi",
    routeTemplate: "api/{controller}/{id}",
    defaults: new { id = RouteParameter.Optional }
);
```

عبارت api، ثابت است و قسمت‌های {controller} و {id} توسط آدرس مقداردهی می‌شوند. زمانی که آدرسی با این الگو تطبیق داشته باشد، کارهای ذیل انجام می‌گیرد:

{controller} به نام Controller تناظر پیدا می‌کند.

نوع درخواست ارسالی (GET, POST, PUT, DELETE) به نام متد تناظر پیدا می‌کند.

اگر قسمت {id} در آدرس وجود داشته باشد، به پارامتر id متد انتخاب شده پاس داده می‌شود.

اگر آدرس دارای Query String باشد، به پارامترهای همنام خود در متد، تناظر پیدا می‌کنند.

در ذیل، مثال هایی را از چند آدرس درخواستی و نتیجه‌ی حاصل از فراخوانی آنها مشاهده می‌کنید.

آدرس /api/products با نوع درخواست GET به متد GetAllProducts()

آدرس /api/products/1 با نوع درخواست GET به متد GetProductById(1)

آدرس /api/products?category=hardware با نوع درخواست GET به متد GetProductByCategory("hardware")

در آدرس اول، عبارت "products" به ProductsController تطبیق پیدا می‌کند. درخواست نیز از نوع GET است، بنابراین Web API به دنبال متدی در Controller می‌گردد که نام آن با عبارت GET "آغاز" شده باشد. همچنین، آدرس شامل قسمت {id} نیز نیست. بنابراین، Web API متدی را انتخاب می‌کند که پارامتر ورودی ندارد. متد GetAllProducts در ProductsController، تمامی این شروط را دارد، پس انتخاب می‌شود.

در دومین آدرس، همان حالت قبل وجود دارد، با این تفاوت که در آدرس درخواستی، قسمت {id} وجود دارد. از آنجا که نوع قسمت {id} در متد GetProductById، int تعریف شده است، باید یک عدد صحیح بعد از آدرس /api/products/ وجود داشته باشد تا متد GetProductById فراخوانی شود. این عدد به طور خودکار به نوع int تبدیل شده و در پارامتر اول متد GetProductById قرار می‌گیرد. در ذیل، برخی آدرس‌ها را ملاحظه می‌کنید که معتبر نیستند و باعث بروز خطا می‌شوند.

آدرس /api/products با نوع درخواست POST، باعث خطای 405Method Not Allowed می‌شود.

آدرس /api/users با نوع درخواست GET، باعث خطای 404Not Found می‌شود.

آدرس /api/products/abc با نوع درخواست GET، باعث خطای 400Bad Request می‌شود.

در آدرس اول، Client یک درخواست از نوع POST ارسال کرده است. Web API به دنبال متدی می‌گردد که نام آن با عبارت Post آغاز می‌شود. اما متدی با این شرط در ProductsController وجود ندارد. بنابراین، پاسخی که دریافت می‌شود، عبارت "405 Method Not Allowed" است. درخواست برای آدرس /api/users/ نیز معتبر نیست، چون Controllerی با نام UsersController وجود ندارد. و سومین آدرس نیز بدین دلیل نامعتبر است که قسمت abc نمی‌تواند به یک عدد صحیح تبدیل شود.

مشاهده‌ی درخواست ارسالی و پاسخ دریافتی

زمانی که با یک وب سرویس کار می‌کنید، مشاهده‌ی محتویات درخواست ارسالی و پاسخ دریافتی می‌تواند کاربرد زیادی در درک نحوه‌ی تعامل بین Client و وب سرویس و کشف خطاهای احتمالی داشته باشد. در Firefox با استفاده از افزونه‌ی Firebug و در Internet Explorer 9 به بالا با ابزار Developer Tools آن می‌توان درخواست‌ها و پاسخ‌ها را مشاهده کرد. در Internet Explorer، کلید F12 را برای اجرای ابزار Developer Tools فشار دهید. از قسمت Network بر روی دکمه‌ی Start Capturing کلیک کنید. حال کلید F5 را برای بارگذاری مجدد صفحه فشار دهید. Internet Explorer، درخواست و پاسخ رد و بدل شده بین مرورگر و Web Server را مانیتور کرده و گزارشی را نشان می‌دهد (شکل ذیل).

F12




File Find Disable View Images Cache Tools Validate | Browser Mode: IE9

HTML CSS Console Script Profiler **Network**

Stop capturing Go to detailed view

URL	Met...	Result	Type	Received	Taken
http://localhost:26481/	GET	200	text/html	1.66 KB	78 ms
/Content/css?v=ji3nO1pdg6VLv3CV...	GET	200	text/css	0.89 KB	47 ms
/Content/themes/base/css?v=UM62...	GET	200	text/css	24.73 KB	63 ms
/Scripts/js?v=4h5lPNUsLiFoa0vqrItj...	GET	200	text/javascript	325.88...	62 ms
/Scripts/jquery-1.6.2.min.js	GET	304	application/x-ja...	176 B	47 ms
/api/products/	GET	200	application/json	398 B	281 ms

از ستون URL، آدرس /api/products/ را انتخاب و بر روی دکمه‌ی Go to detailed view کلیک کنید. در قسمتی که باز می‌شود، گزینه‌هایی برای مشاهده‌ی هدرهای درخواست، پاسخ و همچنین بدنه‌ی هر یک وجود دارد. به عنوان مثال، اگر قسمت Request headers را انتخاب کنید، خواهید دید که Internet Explorer از طریق هدر Accept، تقاضای پاسخ در قالب JSON را کرده است (شکل ذیل).

HTML CSS Console Script Profiler Network	
   Stop capturing Back to summary view < Prev	
URL: http://localhost:26481/api/products/	
Request headers Request body Response headers Response body Cookies Initiator Time	
Key	Value
Request	GET /api/products/ HTTP/1.1
X-Requested-With	XMLHttpRequest
Accept	application/json, text/javascript, */*; q=0.01
Referer	http://localhost:26481/
Accept-Language	en-us
Accept-Encoding	gzip, deflate
User-Agent	Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; Trident/5.0)
Host	localhost:26481
Connection	Keep-Alive

اگر قسمت Response body را انتخاب کنید، پاسخ دریافت شده در قالب JSON را خواهید دید.

در قسمت بعد، با مدیریت کدهای وضعیت HTTP برای اعمال چهارگانه‌ی CRUD آشنا می‌شوید.

نظرات خوانندگان

نویسنده: نیما

تاریخ: ۱۳۹۱/۰۴/۱۹ ۱۲:۱۲

سلام آقای راد

با تشکر از زحمتی که میکشید. فرمودید که :

"بنابراین web api به دنبال متدی در controller می‌گردد که نام آن با عبارت "get" آغاز شده باشد. "

آیا این کار باعث عدم دقت و ایجاد خطاهای ناخواسته نمیشه؟ این فقط متدی با get شروع بشه شاید برای من که خیلی کم mvc کار کردم یکم مشکل دار به نظر برسه. اگر ما دو متد داشته باشیم که در ابتدای آنها get باشد آیا برنامه خطا میگیرد؟ ممنون میشم یکم در این باره توضیح بدین

نویسنده: بهروز راد

تاریخ: ۱۳۹۱/۰۴/۱۹ ۱۲:۲۶

شما محدود به رفتار پیش فرض Web API نیستید. می‌تونید route رو تغییر بدید و نام Action رو هم در اون ذکر کنید.

```
routes.MapHttpRoute(
    name: "DefaultApi",
    routeTemplate: "api/{controller}/{action}/{id}",
    defaults: new { id = RouteParameter.Optional }
);
```

و در ProductsController داشته باشید:

```
[HttpGet]
public string Details(int id)
{
    // do something
}
```

حال درخواستی برای /api/products/details/1 باعث اجرای متد Details میشه. یا حتی می‌تونید route رو تغییر ندید و فقط از [HttpGet] و [HttpPost] و امثال اونها برای تعیین فعل استفاده کنید. به عنوان مثال، اگر route پیش فرض رو تغییر ندید و متد Details رو به شکل قبل داشته باشید، آدرسی مانند /api/products/1 با نوع GET باعث میشه تا متد Details اجرا بشه.

نویسنده: نیما

تاریخ: ۱۳۹۱/۰۴/۱۹ ۱۲:۲۷

بسیار ممنونم خیلی مفید بود

نویسنده: آریا

تاریخ: ۱۳۹۱/۰۴/۱۹ ۱۹:۰۱

خیلی عالی بود. متشکر

نویسنده: رضا ب.
تاریخ: ۱۳۹۱/۰۴/۲۱ ۱۰:۳۰

یه سوال که ربط چندانی به این پست نداره؛
asp.net web api رو میتونیم یه لایه abstraction حساب کنیم که در اون منطق سیستم (BL) وجود داره و بنابراین از آن در سطح انتزاعی بالاتری در سیستم یا سیستم‌های مشابه استفاده میشن؟ (تاکید سوالم آنجاست که میزان عملکرد موثر asp.net web api تا کجاست؟)
ممنون.

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۴/۲۱ ۱۹:۳۸

شما در سواالتون می‌تونید عبارت "ASP.NET Web API" رو با "Web Service تحت HTTP" جایگزین کنید. در Web Service هم منطق سیستم وجود داره، مثلاً محاسبه‌ی نرخ تورم در یک بازه‌ی زمانی با توجه به 30 قلم کالای اساسی. عملکرد Web API، همان عملکردی است که از یک Web Service تحت HTTP مانند ASMX انتظار دارید.

نویسنده: حمزه ء
تاریخ: ۱۳۹۳/۰۲/۲۰ ۱۲:۲۶

در قسمت سوم آموزش این مثال رو داشتیم :

```
$.getJSON("api/products/"+id,
    function (data) {
        var str = data.Name + ' ' + data.Price;
        $('#products').empty();
        $('#products').html(str);
    }
);
```

خب تا اینجا api/products/id اجرا میشه .
فرض کنید چند جستجو داریم و نیاز داریم برای هر کدوم اکشن متناظر با اون اجرا بشه برای مثال:
api/products/id
api/products/details/id
حالا چطور میتونم برای دو دکمه تعیین کنم ، با زدن هر کدوم چه تابعی اجرا بشه ؟
بهتر بگم چطور details رو برای یک دکمه به آدرس اضافه کنم ؟

نویسنده: محسن خان
تاریخ: ۱۳۹۳/۰۲/۲۰ ۱۳:۲

از [متد click](#) استفاده کنید. داخل callback آن درخواست Ajax ایی را ارسال کنید به سرور.

نویسنده: حمزه ء
تاریخ: ۱۳۹۳/۰۲/۲۰ ۱۸:۲

ممنونم .
1- کدها رو رویداد کلیک نوشتیم و اجرا شد . ولی توی آدرس بار مرورگر هیچ تغییری بوجود نیومد ؟ چطور میتونم زمانی که یک متد رو از web api فراخوانی کردم ، همزمان آدرس بار مرورگر هم تغییر کنه ؟
2- برای اینکه فقط یوزرهای سایت و آنلاین شده یا role های خاص بتونن از اون متد استفاده کنن ، attribute رو بالای اون اضافه کردم ، آیا درسته ؟

```
[Authorize(Roles="Admin")]
//[Authorize(Users="")]
```

```
public Product GetProductById(int Id)
{
    var product = Products.FirstOrDefault(p => p.Id == Id);
    if(product==null)
    {
        throw new HttpResponseException(HttpStatusCode.NotFound);
    }
    return product;
}
```

نویسنده:

محسن خان

تاریخ:

۱۸:۵۸ ۱۳۹۳/۰۲/۲۰

این ASP.NET MVC نیست. ASP.NET Web API است. می‌تونی دستی آدرس خاصی رو در مرورگر وارد کنی و نهایتاً مثلاً خروجی JSON یا XML بگیری (شاید بهتر باشه یکبار اینکار رو انجام بدی تا حس بهتری نسبت به این فناوری پیدا کنی که کارش چی هست. خروجی‌اش چی هست). در کل هدفش این نیست که خروجی HTML به شما بده. هدفش تامین داده برای کلاینت‌ها هست. سمت کلاینت رو آزاد هستی هر طور که دوست داشتی کار کنی. مثلاً یک صفحه‌ی HTML درست کنی و اطلاعات Web API رو بگیری و نمایش بدی.