

آموزش MVC Core-ASP.NET Core 2.0 Response Caching

Problem

نحوه ذخیره (cache) responses در ASP.NET Core.

Solution

برای یک پروژه خالی ایجاد کنید، services و middleware برای ذخیره در Startup پیگیربندی کنید،

```
public void ConfigureServices(  
    IServiceCollection services)  
{  
    services.AddResponseCaching();  
    services.AddMvc();  
}  
  
public void Configure(  
    IApplicationBuilder app,  
    IHostingEnvironment env)  
{  
    app.UseResponseCaching();  
    app.UseMvcWithDefaultRoute();  
}
```

یک کنترلر ایجاد کنید، و ویژگی [ResponseCache] را اضافه کنید:

```
[ResponseCache(Duration = 60)]  
public class HomeController : Controller  
{  
    public IActionResult Index()  
    {  
        return View(new HomeOutputModel  
    {
```

```
LastUpdated = DateTime.Now
});
}
}
```

پاسخ حاوی یک هدر Cache-Control است.

```
Cache-Control: public,max-age=60
```

همانطور که بین کنترلر با قابلیت ذخیره سازی (Home) (caching) و یکی دیگر بدون آن (Movies) حرکت می کنید، متوجه خواهید شد که زمان به روزرسانی نشده است، یعنی آن را از سرور دریافت نمی کند بلکه از حافظه پنهان دریافت می کند.

Home

Last Updated: 18/07/2017 15:45:13

[Movies](#)

Movies

Last Updated: 18/07/2017 15:46:13

[Home](#)

Home

Last Updated: 18/07/2017 15:45:13

[Movies](#)

Discussion

Response Caching middleware مسئول ذخیره و خدمت به پاسخ از cache است. هدرهای مرتبط با حافظه پنهان را به پاسخ HTTP اضافه می کند، و Cache-Control اصلی آن است.

هنگامی که middleware تنظیم می شود، ویژگی [ResponseCache] به action / controller اعمال می شود تا caching را فعال کند. این ویژگی دارای برخی از پارامترهای مفید برای تغییر رفتار middleware است.

- Duration : استفاده از تنظیم زمان کش (در ثانیه).
- Location : به هدر Cache-Control به عنوان private، public یا no-cache ترجمه می شود.

- VaryByQueryKeys: پاسخ ها بر اساس پارامترهای query string ذخیره می شوند.
- VaryByHeader: استفاده می شود تا هدر پاسخ Vary HTTP را تنظیم کند.
- CacheProfileName: میتواند به تنظیمات پروفایل حافظه پنهان در middleware MVC اشاره کند (نگاه کنید به زیر).

```
services.AddMvc(options =>
{
    options.CacheProfiles.Add("default", new CacheProfile
    {
        Duration = 30,
        Location = ResponseCacheLocation.Any
    });
});
```

Logging

شما می توانید ورود به سیستم را فعال کنید تا عملکرد میان افزار Response Caching middleware را مشاهده کنید. در زیر اولین درخواست برای Index / Home است (خطوط highlighted را ببینید). درخواست های بعدی در این چرخه عمر (lifecycle) قرار می گیرند و به صفحه از حافظه پنهان (تا تاریخ انقضای آن) خدمت می کنند:

```
Info: Microsoft.AspNetCore.Hosting.Internal.WebHost[1]
      Request starting HTTP/1.1 GET http://localhost:58513/home
Info: Microsoft.AspNetCore.ResponseCaching.ResponseCachingMiddleware[24]
      No cached response available for this request.
Info: Microsoft.AspNetCore.Mvc.Internal.ControllerActionInvoker[1]
      Executing action method Fiver.ResponseCaching.Controllers.HomeController.Index
      (Fiver.ResponseCaching.Controllers.HomeController) - ModelState is Valid
Info: Microsoft.AspNetCore.Mvc.ViewFeatures.Internal.ViewResultExecutor[1]
      Executing ViewResult, running view at path /Views/Home/Index.cshtml.
Info: Microsoft.AspNetCore.Mvc.Internal.ControllerActionInvoker[2]
      Executed action Fiver.ResponseCaching.Controllers.HomeController.Index (Fiver.R
Info: Microsoft.AspNetCore.ResponseCaching.ResponseCachingMiddleware[26]
      The response has been cached.
Info: Microsoft.AspNetCore.Hosting.Internal.WebHost[2]
      Request finished in 3.8534ms 200 text/html; charset=utf-8
```

source code را می توانید از لینک زیر دانلود کنید:

<https://github.com/TahirNaushad/Fiver.Perf.ResponseCaching>