

قسمت چهارم

15. استفاده از using

اگر از objectهایی استفاده می‌کنید که interface مربوط به [IDisposable](#) را پیاده سازی کرده اند، حتما از عبارت using استفاده کنید. استفاده از دستور using باعث می‌شود زمانی که دیگر نیازی به object شما نباشد، به صورت خودکار از حافظه حذف شود و در روال جمع آوری زباله (GC) قرار گیرد. این عمل باعث حداقل رسیدن احتمال نشت حافظه در نرم افزار شما می‌شود. برای مثال:

```
using System;
using System.Text;

class Program
{
    static void Main()
    {
        // Use using statement with class that implements Dispose.
        using (SystemResource resource = new SystemResource())
        {
            Console.WriteLine(1);
        }
        Console.WriteLine(2);
    }
}

class SystemResource : IDisposable
{
    public void Dispose()
    {
        // The implementation of this method not described here.
        // ... For now, just report the call.
        Console.WriteLine(0);
    }
}
```

برای اطلاعات بیشتر می‌توانید از [این مقاله](#) استفاده کنید.

16. اطلاعات ارسالی توسط شبکه را به حداقل برسانید

حجم اطلاعات ارسالی به شبکه را به حداقل برسانید. ارسال اطلاعات در شبکه به معنی گذر اطلاعات شما از 7 لایه مختلف شبکه در رایانه شما، گذر از media شبکه، گذر مجدد از 7 لایه شبکه در رایانه مقصد می‌باشد. به این معنی که هرچه اطلاعات بیشتری در شبکه ارسال کنید، سربار بیشتری متوجه سیستم شما خواهد بود. برای رفع این مشکل از فشرده سازهای css و javascript استفاده کنید. این فشرده سازها فواصل خالی، دستورات اضافی و... را از کد شما حذف و حجم آن را به حداقل می‌رسانند. کم کردن تعداد درخواست‌ها و در نتیجه کم کردن تعداد فایل‌های ارسالی از سرور به کاربر نیز حربه ای در این زمینه می‌باشد. برای مقایسه فشرده سازها به صورت آنلاین و استفاده از بهترین آنها (متناسب کد شما) می‌توانید از [این سایت](#) استفاده کنید. امکانات توکاری هم وجود دارد که در زمان اجرای برنامه css و javascript شما را فشرده و تلفیق می‌کند ولی با توجه به اینکه برای سرور در هر مرتبه فراخوانی سربار دارد (حتی در صورت کش کردن) اکیدا توصیه می‌شود از فشرده سازها قبل از اجرای برنامه (Pre-Compressed) به جای فشرده سازهای زمان اجرا (Run-time Compression) استفاده کنید.

نظرات خوانندگان

نویسنده:

حسین حقیان

تاریخ:

۱۳۹۲/۰۵/۱۶ ۱۲:۳۸

با سلام و تشکر از مجموعه مطالب مرتبط که ارائه کردید

میشه برای این قسمت مثالی رو ذکر بفرمایید

اکیدا توصیه می‌شود از فشرده سازها قبل از اجرای برنامه (Pre-Compressed) به جای فشرده سازهای زمان اجرا (Run-time Compression) استفاده کنید.

با تشکر

نویسنده:

م منفرد

تاریخ:

۱۳۹۲/۰۵/۱۶ ۱۹:۵۱

فشرده سازی قبل از اجرای برنامه (Pre-Compression) یعنی که شما قبل از اینکه برنامه خود را در محیط اصلی نصب و اجرا کنید، فایل‌های اسکریپت آن را فشرده کنید. یعنی کاربران فایل اسکریپت فشرده شده را درخواست و دانلود می‌کنند و عملیات اضافی در سمت سرور انجام نمی‌شود. به عنوان مثال شما از فایل JQuery.min.js به جای jquery.js استفاده کنید. یعنی استفاده از نسخه فشرده شده اسکریپت‌ها.

فشرده سازی زمان اجرا (Run-time Compression) یعنی فشرده سازی اسکریپت‌های مورد نیاز کاربر توسط خود برنامه وب (به صورت خودکار و یا توسط یک ماژول اضافی). این عمل باعث می‌شود که در هر بار درخواست هر کاربر برای یک فایل، برنامه آن را مجدد فشرده سازی کند (و یا از cache استفاده کند). این عمل به معنی استفاده بیشتر از منابع پر ارزش سرور شما می‌باشد. به عنوان مثال شما بخواهید در هر مرحله درخواست هر کاربر jquery.js را فشرده کنید!

از مقایسه دو حالت بالا مشخص است وقتی شما فقط یکبار اسکریپت‌های خود را فشرده می‌کنید بسیار از حالتی که در هر مرتبه از درخواست کاربران آن را فشرده کنید بهتر است و کمتر منابع سرور را هدر می‌دهد.