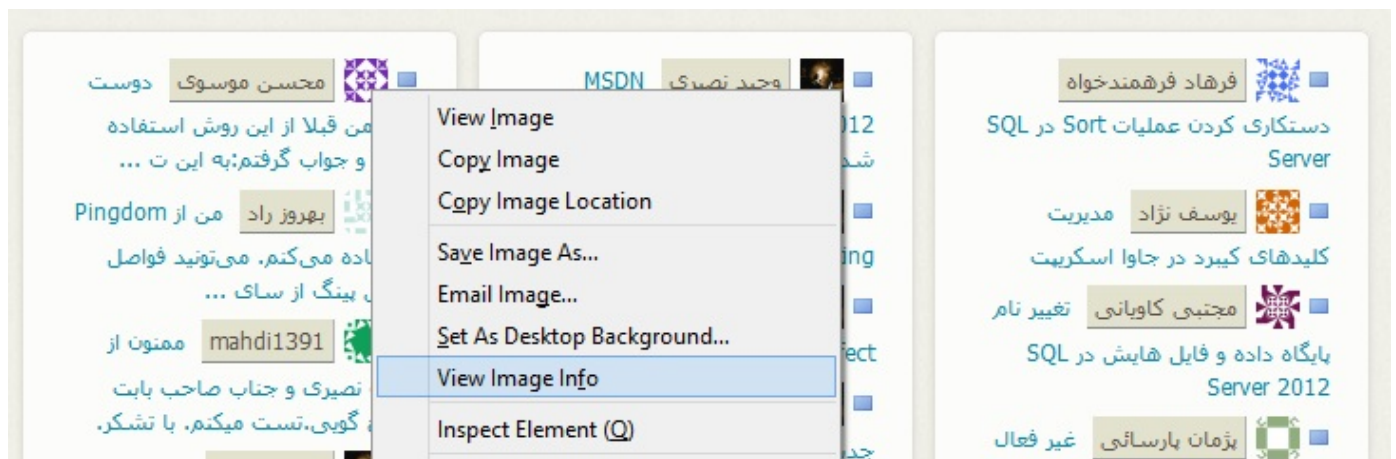


برای برنامه نویسان همیشه این امکان هست که تصاویری را که از کاربر دریافت می‌کنند تغییر اندازه دهند، مثلاً در همین سایت تصاویری از کاربران جهت نمایش در پروفایل آنها دریافت می‌شود، در همین سایت نیز این اتفاق می‌افتد مثلاً تصاویر پروفایل کاربران با اندازه‌های متفاوتی نشان داده می‌شود.



برای انجام این کارها میتوان به دو طریق عمل کرد:

تغییر اندازه تصویر در زمان ذخیره‌سازی

در زمانی که می‌خواهیم تصویر را به بازدید کننده نشان دهیم

در حالت 1 زمانی که تصویری را از کاربر دریافت می‌کنیم با توجه به اینکه تصویر را با چه اندازه‌هایی در نرم افزار نیاز داریم تغییر اندازه داده و تک تک ذخیره می‌کنیم، این روش کل عملیات در زمان ثبت و تنها یکبار اتفاق می‌افتد، این روش جای بیشتری از منابع (مانند هارد دیسک یا دیتابیس) سرور را اشغال می‌کند اما در عوض می‌توان گفت سرعت بالاتری دارد، در روش دوم زمانی که بازدید کننده از سایت (نرم افزار) بازدید می‌کند تصویر اصلی با توجه به نیاز تغییر اندازه داده شده و برای کاربر ارسال می‌شود (در واقع کاربر آن را مشاهده می‌کند)، این روش فضای کمتری از منابع را اشغال می‌کند اما در زمان اجرا عملیات اضافی برای هر کاربری (البته با کش کردن این عملیات کم می‌شود) انجام می‌شود.

در هر دو روش گفته شده در هر صورت ما باید متد (توابعی) برای تغییر اندازه تصویر داشته باشیم که در زیر نحوه نوشتن آن را شرح خواهیم داد.

```
using System;
using System.Drawing;
using System.Drawing.Drawing2D;
using System.Drawing.Imaging;
using System.IO;

namespace PWS
{
    public static class Helpers
    {
        /// <summary>
        /// تغییر اندازه تصویر
        /// </summary>
        /// <param name="imageFile">آرایه بایتی از تصویر مورد نظر</param>
        /// <param name="targetSize">اندازه تصویر خروجی</param>
        /// <param name="format">فرمت تصویر خروجی</param>
    }
}
```

```

    /// <returns></returns>
    public static byte[] ResizeImageFile(this byte[] imageFile, Int32 targetSize, ImageFormat
format)
    {
        if (imageFile == null)
            throw new Exception("لطفا تصویر اصلی را مشخص نمایید");
        // باز کردن تصویر اصلی به عنوان یک جریان
        using (var oldImage = Image.FromStream(new MemoryStream(imageFile)))
        {
            // محاسبه اندازه تصویر خروجی با توجه به اندازه داده شده
            var newSize = CalculateDimensions(oldImage.Size, targetSize);
            // ایجاد تصویر جدید
            using (var newImage = new Bitmap(newSize.Width, newSize.Height,
PixelFormat.Format24bppRgb))
            {
                using (var canvas = Graphics.FromImage(newImage))
                {
                    canvas.SmoothingMode = SmoothingMode.AntiAlias;
                    canvas.InterpolationMode = InterpolationMode.HighQualityBicubic;
                    canvas.PixelOffsetMode = PixelOffsetMode.HighQuality;
                    // تغییر اندازه تصویر اصلی و قرار دادن آن در تصویر جدید
                    canvas.DrawImage(oldImage, new Rectangle(new Point(0, 0), newSize));
                    var m = new MemoryStream();
                    // ذخیره تصویر جدید با فرمت وارد شده
                    newImage.Save(m, format);
                    return m.GetBuffer();
                }
            }
        }
    }

    /// <summary>
    /// محاسبه ابعاد تصویر خروجی
    /// </summary>
    /// <param name="oldSize">اندازه تصویر اصلی</param>
    /// <param name="targetSize">اندازه تصویر خروجی</param>
    /// <returns></returns>
    private static Size CalculateDimensions(Size oldSize, Int32 targetSize)
    {
        var newSize = new Size();
        if (oldSize.Height > oldSize.Width)
        {
            newSize.Width = Convert.ToInt32(oldSize.Width * (targetSize / (float)oldSize.Height));
            newSize.Height = targetSize;
        }
        else
        {
            newSize.Width = targetSize;
            newSize.Height = Convert.ToInt32(oldSize.Height * (targetSize / (float)oldSize.Width));
        }
        return newSize;
    }
}

```

در متد `ResizeImageFile` تصویر اصلی به عنوان یک جریان باز می‌شود. (سطر 23)

اندازه تصویر خروجی با توجه به اندازه وارد شده توسط متد `CalculateDimensions` تعیین می‌شود؛ روال کار متد `CalculateDimensions` اینگونه است که اندازه عرض و ارتفاع تصویر اصلی بررسی می‌شود و با توجه به اینکه کدام یک از اینها بزرگتر است تغییر اندازه در آن صورت می‌گیرد، مثلا در صورتی که عکس ارتفاع بیشتری نسبت به عرض تصویر داشته باشد تصویر تغییر اندازه داده شده نیز با توجه به تناسب ارتفاع تغییر داده می‌شود و بالعکس. (سطر 26)

پس از تغییر اندازه تصویر جدیدی در حافظه ایجاد می‌شود. (سطر 28)

سپس تنظیمات گرافیکی لازم بروی تصویر جدید اعمال می‌شود. (سطر 30 تا 34)

تصویر اصلی با توجه به اندازه جدید تغییر کرده و در تصویر جدید قرار می‌گیرد. (سطر 36)

در نهایت تصویر جدید با فرمت وارد شده در متد ذخیره شده و به عنوان خروجی متد برگشت داده می‌شود. (سطر 37 تا 40)

خروجی این متد نیز آرایه بایتی می‌باشد که به سادگی می‌توانید از آن برای ذخیره تصاویر در دیتابیس استفاده نمایید.

نحوه استفاده از این تابع در ASP.NET می‌تواند به صورت زیر باشد :

```
byte[] oldImage = FileUploadImage.FileBytes;  
byte[] target = oldImage.ResizeImageFile(100, ImageFormat.Jpeg);
```

در واقع فراخوانی مذکور تصویر ورودی را به اندازه 100 پیکسل تغییر داده و در آرایه بایتی به نام target ذخیره می‌کند. در بخش بعد در زمان نمایش تصویر به کاربر آن را تغییر اندازه خواهیم داد. لازم به ذکر است که کدهای تغییر اندازه از StarterKit های میکروسافت کپی برداری شده است.

نظرات خوانندگان

نویسنده: محسن.د
تاریخ: ۱۳:۲۴ ۱۳۹۱/۰۹/۱۴

بسیار عالی

در asp.net MVC کلاس [webImage](#) در فضا نام [System.Web.Helpers](#) این امکانات رو در اختیار توسعه دهنده قرار میده .

نویسنده: صابر فتح الهی
تاریخ: ۱۵:۰ ۱۳۹۱/۰۹/۱۴

بله دیدم جالب بود واسم، اما تغییر اندازه ندیدم کدوم متده؟

نویسنده: محسن.د
تاریخ: ۱۸:۴ ۱۳۹۱/۰۹/۱۴

متد [WebImage.Resize](#)

نویسنده: پ.م
تاریخ: ۱۴:۳۶ ۱۳۹۳/۰۲/۲۷

لطفا در مورد اینکه در زمان واکنشی تصاویر از دیتابیس (کش کنیم) توضیح دهید
چون خیلی سرعت کار با سایت پایین میاد
مثلا هر بار کاربر که بخواد از صفحه ای به صفحه ای دیگه بره تصاویر از نو بارگزاری میشن و این سرعت سایت رو کند میکنه

نویسنده: وحید نصیری
تاریخ: ۱۴:۳۹ ۱۳۹۳/۰۲/۲۷

به مطلب و مآخذ « [مدیریت درخواستهای شرطی در ASP.NET MVC](#) » مراجعه کنید. نکات کلی در بحث عنوان شده. در مآخذ آن پیاده سازیهای مخصوص وب فرمها هم هست.

نویسنده: رضایی
تاریخ: ۱۹:۱۷ ۱۳۹۳/۰۵/۳۱

سلام؛ موقع تغییر سایز یک تصویر png به png پس زمینه رو مشکلی میکنه. آیا راهی هست در صورتی که تصویر اولیه png باشه تصویر نهایی هم بدون پس زمینه باشه.

نویسنده: وحید نصیری
تاریخ: ۱۳:۵ ۱۳۹۳/۰۶/۲۰

[Resize PNG Image and Keep Transparent Background](#)