# How to resize image after being uploaded in ASP.NET Core

## Introduction

This sample demonstrates how to resize image after being uploaded in ASP.NET Core.

## Sample prerequisites

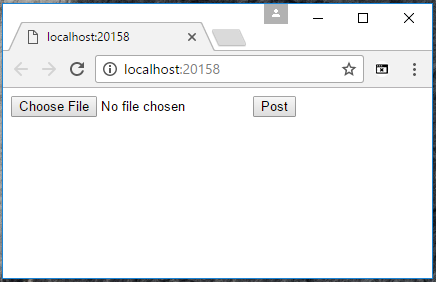
* .NET Core 1.0 or later version(s). [[.NET Core + Visual Studio tooling](http://go.microsoft.com/fwlink/?LinkID=798306)]
* Microsoft Visual Studio 2015 update3 or above. [[Visual Studio 2015](https://www.visualstudio.com/en-us/visual-studio-homepage-vs.aspx)]

## Building the sample

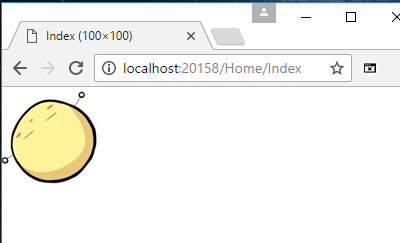
* Open the sample solution “**ResizeImageASPNETCore.sln**” using Visual Studio.
* Right click on project “**ResizeImageASPNETCore**” and select Restore packages.
* Press **F6 Key** or select **Build -> Build Solution** from menu to build this solution.

## Running the sample

* Open the sample solution using Visual Studio, then press **F5 Key** or select **Debug -> Start Debugging** from the menu.
* When application is running, you can see below UI.



* Choose an image file and click post button, if the image size greater than 100 \* 100, the server will resize this image and show you.



## Using the code

Extends.cs

public static Image Resize(this Image current, int maxWidth, int maxHeight)

{

int width, height;

#region reckon size

if (current.Width > current.Height)

{

width = maxWidth;

height = Convert.ToInt32(current.Height \* maxHeight / (double)current.Width);

}

else

{

width = Convert.ToInt32(current.Width \* maxWidth / (double)current.Height);

height = maxHeight;

}

#endregion

#region get resized bitmap

var canvas = new Bitmap(width, height);

using (var graphics = Graphics.FromImage(canvas))

{

graphics.CompositingQuality = CompositingQuality.HighSpeed;

graphics.InterpolationMode = InterpolationMode.HighQualityBicubic;

graphics.CompositingMode = CompositingMode.SourceCopy;

graphics.DrawImage(current, 0, 0, width, height);

}

return canvas;

#endregion

}

public static byte[] ToByteArray(this Image current)

{

using (var stream = new MemoryStream())

{

current.Save(stream, current.RawFormat);

return stream.ToArray();

}

}

}

HomeController.cs

public IActionResult Index()

{

return View();

}

[HttpPost]

public FileStreamResult Index(IList<IFormFile> files)

{

using (Image img = Image.FromStream(files[0].OpenReadStream()))

{

Stream ms = new MemoryStream(img.Resize(100, 100).ToByteArray());

return new FileStreamResult(ms, "image/jpg");

}

}

## More information

Bitmap in MSDN

<https://msdn.microsoft.com/en-us/library/system.drawing.bitmap(v=vs.110).aspx>