Exercise

Start by designing the public API (DI style)

Download images from web (limit the download rate to the processing capacity)



Transform each image on multiple channels, each using different image effect.



Save the processed imaged int a storage.



Take into account that the processing can be delegate to the cloud.

- Image processing libraries: see on the next pages
- Image sources:
 - https://source.unsplash.com/random/1200x1200
 - https://dog.ceo/api/breeds/image/random

Bonus: before saving the images, combine the original image with all of it's manipulation into single image.

Exercise (image providers)

```
Sample for SkiaSharp API
NuGet: SkiaSharp
Doc: <a href="https://developer.xamarin.com/api/namespace/SkiaSharp/">https://developer.xamarin.com/api/namespace/SkiaSharp/</a>
using(var fs = File.OpenRead(path))
using (var stream = new SKManagedStream(fs))
using (var dest = File.OpenWrite("target.jpg"))
using (var bitmap = SKBitmap.Decode(stream))
using (var filter = SKImageFilter.CreateBlur(50, 50))
using (var surface = SKSurface.Create(bitmap.Info))
using (var paint = new SKPaint())
    paint.ImageFilter = filter;
    SKCanvas canvas = surface.Canvas;
    canvas.DrawBitmap(bitmap, SKRect.Create(bitmap.Width, bitmap.Height), paint);
    SKData data = surface.Snapshot().Encode(SKEncodedImageFormat.Jpeg, 80);
    data.SaveTo(dest);
```

https://docs.microsoft.com/en-us/xamarin/graphics-games/skiasharp/introduction https://skia.org/