

Top 5

Image Optimization Techniques

.JPG



.WEBP



1. WebP Format

A modern image format that provides both lossy and lossless compression. WebP images can be up to 30% smaller than JPEGs and PNGs, significantly reducing page load times.

```
  
  
<!--Better in every way!-->  

```

The `` tag directly uses the WebP file as the source since all modern browsers support this format.

2. Srcset and sizes

Provide different image resolutions for different screen sizes. Improves load times by delivering images tailored to the user's device and screen resolution.

```

```

The browser selects the best image size based on the device's screen width and resolution, optimizing load times.

3. Image Lazy Loading

A technique that delays the loading of images until they are about to enter the viewport. Reduces initial page load time and conserves bandwidth by loading images only when needed.

```

```

The `loading="lazy"` attribute is natively supported by modern browsers and defers image loading until it's needed.

4. Using SVGs

SVGs scale to any size without losing quality. They are perfect for icons, logos, and illustrations. They're also lightweight, making them ideal to increase the performance of your website.

```
<svg width="100" height="100" xmlns="http://www.w3.org/2000/svg">  
  <circle  
    cx="50"  
    cy="50"  
    r="40"  
    stroke="black"  
    stroke-width="3"  
    fill="red" />  
</svg>
```

The SVG code is directly embedded in the HTML, allowing it to scale without losing quality and load faster than raster images.

5. Image Compression

Compression reduces image file size, making them faster to load. Lossy compression sacrifices some quality for smaller sizes, while lossless compression retains all the original data.

Tools you can use to accomplish this:

TinyPNG or ImageOptim

NodeJs Library: Sharp

Photoshop