



Images on the Web

HTML course: Lesson 7

Lesson Plan

1

Different image formats

2

Methods of inserting images into a website

3

Working with decorative properties

4

Optimization

Formats

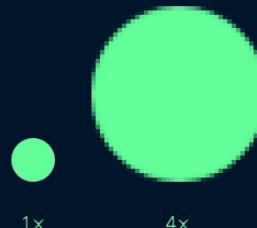


Image Formats

Raster Images

- Consist of pixels
- Have a fixed resolution

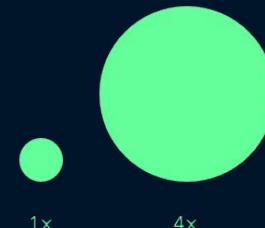
- JPEG
- PNG
- GIF
- WEBP
- AVIF



Vector Images:

- Created using geometric **shapes**, such as **lines and curves**

- SVG



VS

Raster	Vector
Consist of pixels	Consist of mathematical formulas, shapes, and lines
Large file size	Very small file size
Scales with quality loss	Looks the same at any size
Can only be converted to other raster formats	Can be converted to raster images
Cannot be modified easily	Can be easily modified (e.g., splitting into components, animating)
Ideal for photographs	Ideal for icons



Methods of Inserting

Raster Images

- HTML
 - **img**
 - **src**
 - **alt**
 - **width**
 - **height**
 - **picture**
 - **figure**
 - **figcaption**
- CSS
 - **background-image**



```
#app #app--background #app--background-image { vYpXNJd?view  
background-image: url(  
    https://images.unsplash.com/photo-14837286423... );  
height: 100%;  
transition: > filter 250ms, transform 250ms;  
width: 100%;  
  
.background-image {  
background-position: > center;  
background-repeat: > no-repeat;  
background-size: cover;
```

```
<figure class="wiggly wiggle2">  
      
    <figcaption>Wassily Kandinsky: Park of St Cloud,  
1906</figcaption>  
</figure>
```



*Wassily Kandinsky: Park of St
Cloud, 1906*

Image insertion





```

```

File naming:

- Use only English words
- No spaces
- No capital letters
- Reflect the location or subject of the image

blue triangle.jpg

BigPicture.jpg

Attributes:

- **width/height**: Specifies dimensions of the image
- Used for displaying images before styles load, in reading mode, and for accessibility

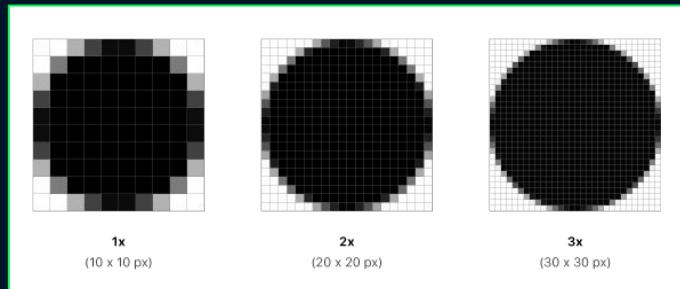
cute-kitten.jpg
feature-block-1.jpg

srcset by pixel destiny

```
<img  
  width="100"  
  height="100"  
  alt="cute corgi"  
  srcset="  
    small-corgi.png 1x,  
    medium-corgi.png 2x,  
    large-corgi.png 3x  
  "  
>
```



1x 2x 3x



Paths

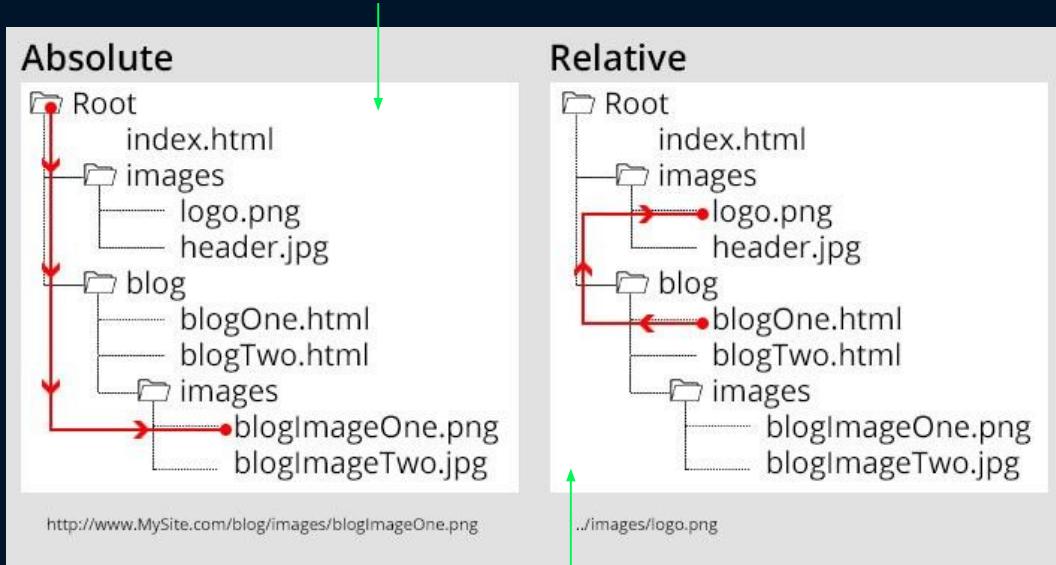
- **Absolute**

- starts with slash
- applicable to server

- **Relative**

- starts with filename or dot and slash ./ or two dots and slash ../
- applicable to file system

`src="https://MySite.com/blog/images/blogImageOne.png"`
`src="/blog/images/blogImageOne.png"`



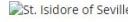
`src=" ../../images/logo.png"`

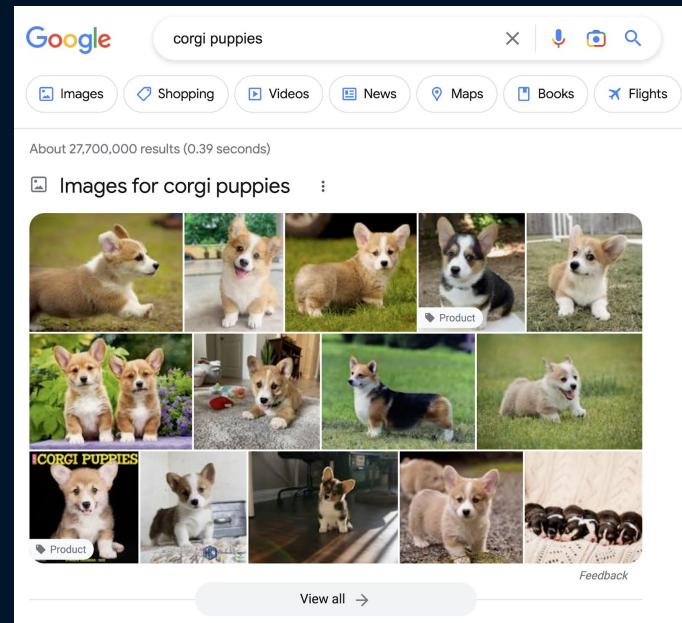
Importance of the `alt` Attribute

```

```

- Screen readers
- Search engines
- Broken images

Digital image that loaded correctly	Digital image that did not load correctly and does not have alt text	Digital image that did not load correctly and does have alt text
		



When alt is not needed

```

```

- Decorative and stock images
- Icons
- When the image caption matches the alt text



A screenshot of a Jimdo website builder interface. It shows a page with three service categories: "BIRTHDAYS" (represented by balloons), "WEDDINGS" (represented by two glasses), and "MUSIC" (represented by musical notes). Each category has a small explanatory text block below it. A large red 'X' is placed above the "MUSIC" icon.

A screenshot of a contact form. It includes a question asking for a phone number, a "Description (optional)" field, and a dropdown for country selection. To the right of the form is a yellow rotary phone on a yellow background, with a large red 'X' placed over the entire area.



- Man wearing a helmet riding a bicycle
- bicycle
- Image of someone riding a bicycle



- Lay's Classic Potato Chips, 1.5 Ounce
- Potato Chips
- buy potato chips online



- Free SEO mini course for beginners by reliablesoft academy.
- download now free
- register to get free course

Good alt Practices

- Summarize what's in the image
- Describe as accurately as possible
- Avoid declaration like "image", "photo", "picture"
- Use the context of the surrounding content
- Avoid repeating text that is in the content
- Don't use very long descriptions, use other attributes for detailed descriptions
- Write in the site's language



Picture insertion



picture: source by type

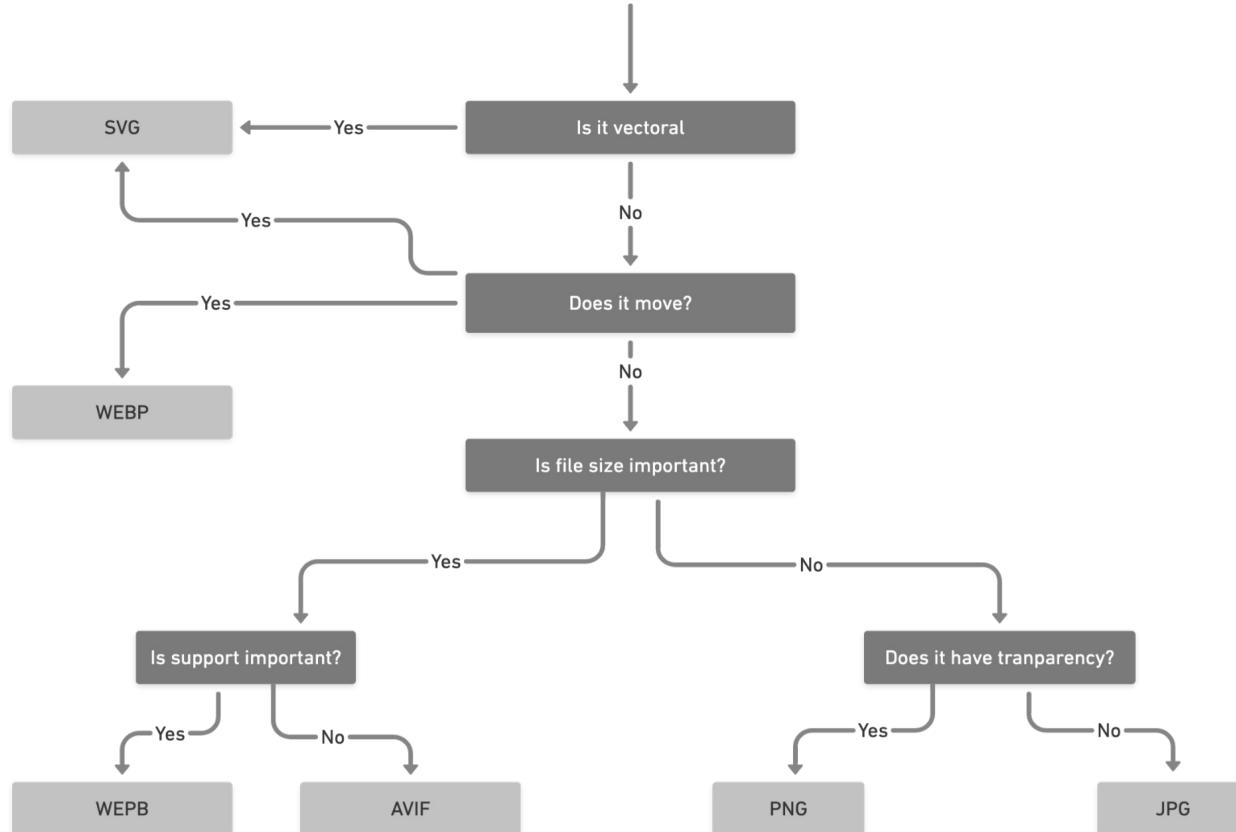


```
<picture>  
  <source srcset="corgi.avif" type="image/avif">  
  <source srcset="corgi.webp" type="image/webp">  
    
</picture>
```

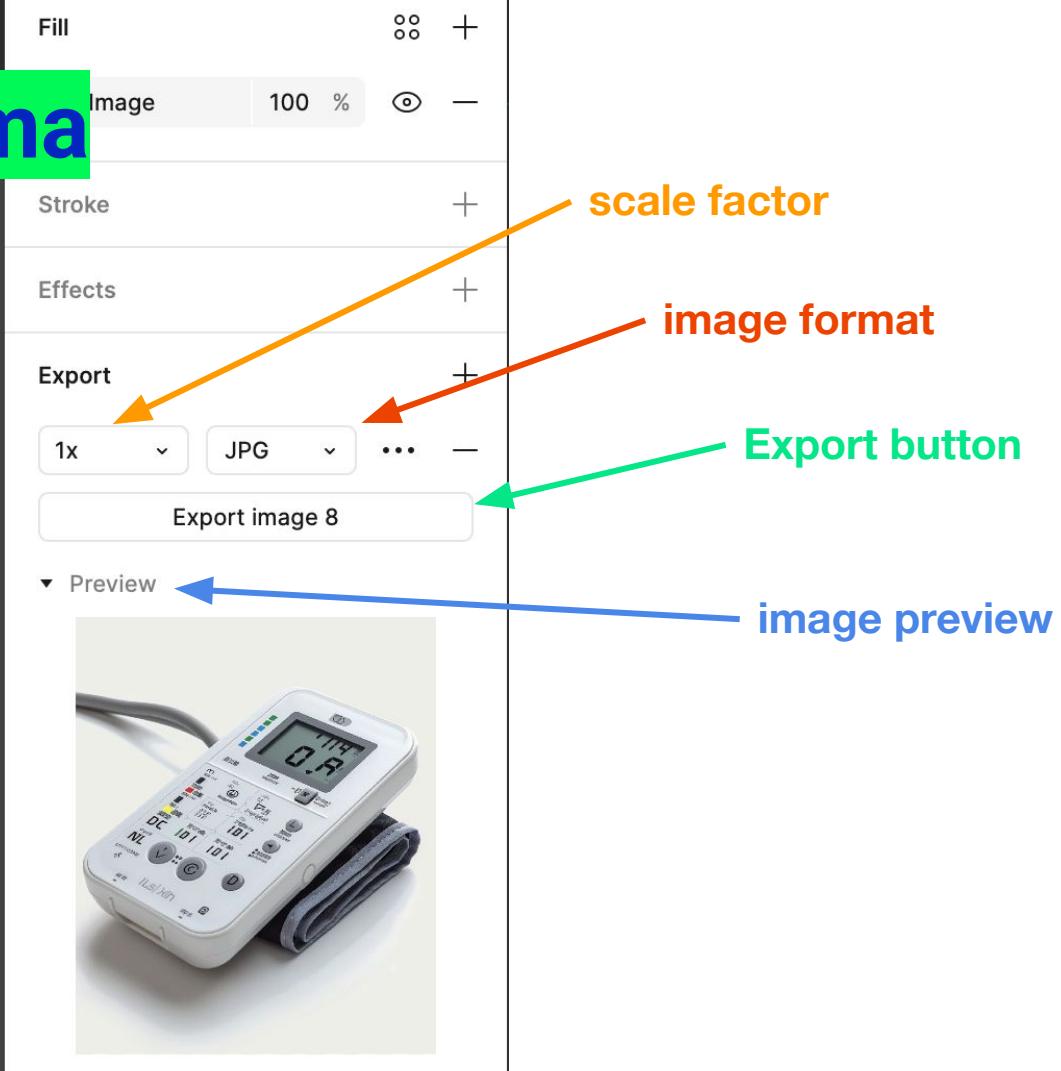
Choosing format



What image format should you pick for your project?



Export from Figma



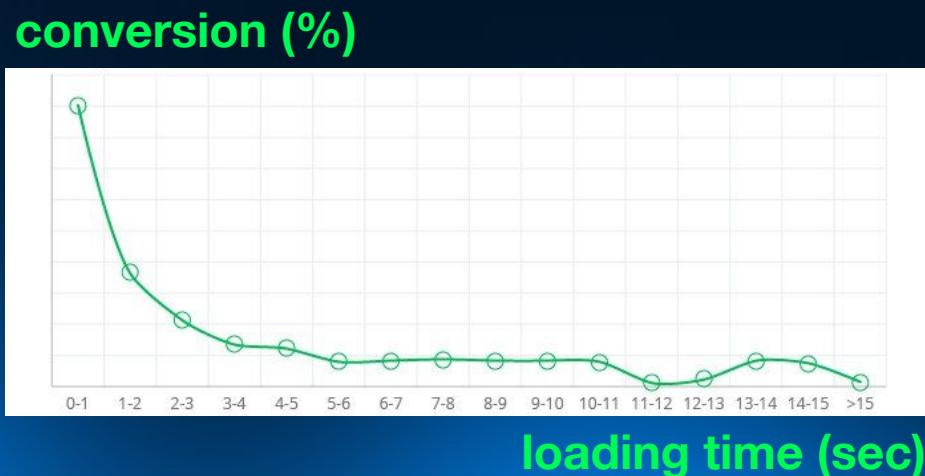
Break time



Optimize



Why Optimize Images



- Improves website loading speed
- Saves traffic and server resources
- Enhances website ranking in search results
- Increases conversion rates

Network

The screenshot shows the Network tab in the Chrome DevTools. The tab bar at the top has several tabs: Elements, Console, Sources, Network (which is selected), Performance, Memory, Application, and more. Below the tabs are various controls like Preserve log, Disable cache, and throttling settings. A large green arrow points from the title 'Network' down to the table below. Another green arrow points from the 'Waterfall' column header down to the waterfall chart area. The main content is a table with columns: Name, Status, Domain, Type, Initiator, Size, Time, and Waterfall. The table lists several resources: logo.svg, image-optimization-shop-cover..., image-optimization-shop-cover..., image-optimization-header.png, shadow.png, and tablet_left.svg. Each row includes a preview icon, status code (200), domain (www.s...), type (e.g., svg+xml, png, avif), initiator (modern-im or non-critical), size, time, and a small waterfall chart. At the bottom, summary statistics are displayed: 6 / 32 requests, 61.3 kB / 1.0 MB transferred, 68.6 kB / 1.6 MB resources, Finish: 472 ms, DOMContentLoaded: 251 ms, and Load: 329 ms.

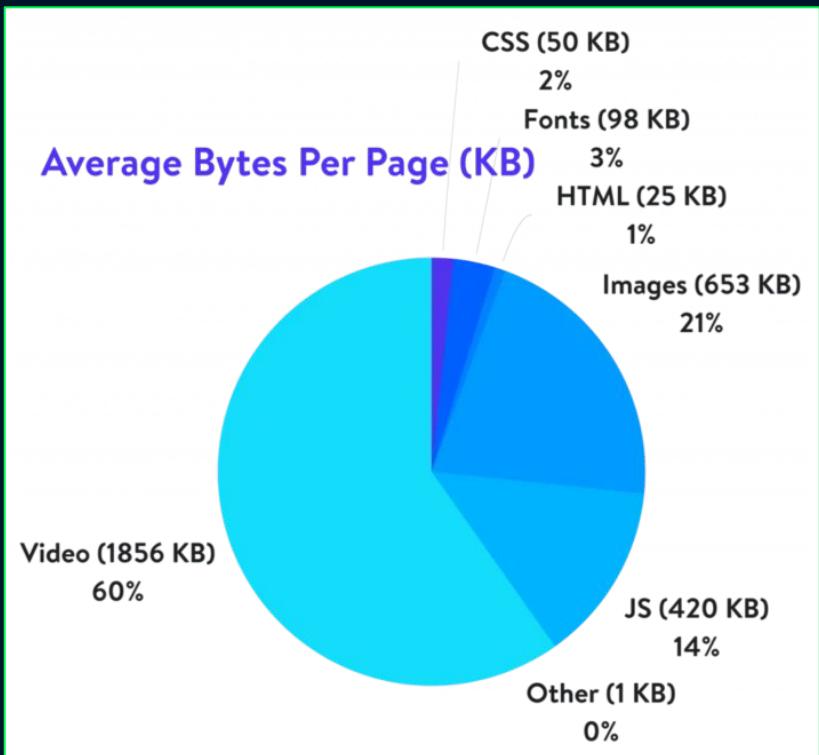
Name	Status	Domain	Type	Initiator	Size	Time	Waterfall
logo.svg	200	www.s...	svg+xml	modern-im	1.2 kB	73 ms	
image-optimization-shop-cover...	200	archive...	png	modern-im	8.9 kB	80 ms	
image-optimization-shop-cover...	200	archive...	avif	modern-im	5.2 kB	91 ms	
image-optimization-header.png	200	archive...	png	modern-im	40.2 kB	102 ms	
shadow.png	200	www.s...	png	non-critical	879 B	72 ms	
tablet_left.svg	200	www.s...	svg+xml	non-critical	4.8 kB	72 ms	

6 / 32 requests | 61.3 kB / 1.0 MB transferred | 68.6 kB / 1.6 MB resources | Finish: 472 ms | DOMContentLoaded: 251 ms | Load: 329 ms

Optimization

Methods

- Defer offscreen images
- Reduce file size
- Crop to exact size
- Change format
- Use CSS styling instead of image merging
- Decomposition
- Using patterns



Optimization methods

Defer offscreen images

```

```

load:

eager tells the browser to load the image as soon as the `` element is processed

lazy tells the browser to load the image shortly before it will appear in the viewport

decoding:

async tells the browser to decode the image asynchronously and allow other content to be rendered before this completes

sync tells the browser to decode the image synchronously for atomic presentation with other content.

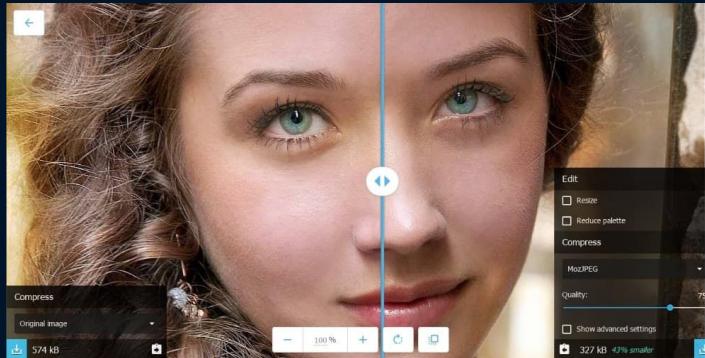
auto tells the browser to decide how to decode

Optimization methods

Reduce File Size

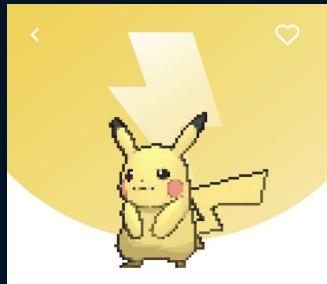
For example

- [squoosh.app](#)
- [SVGOMG](#)



Optimization methods

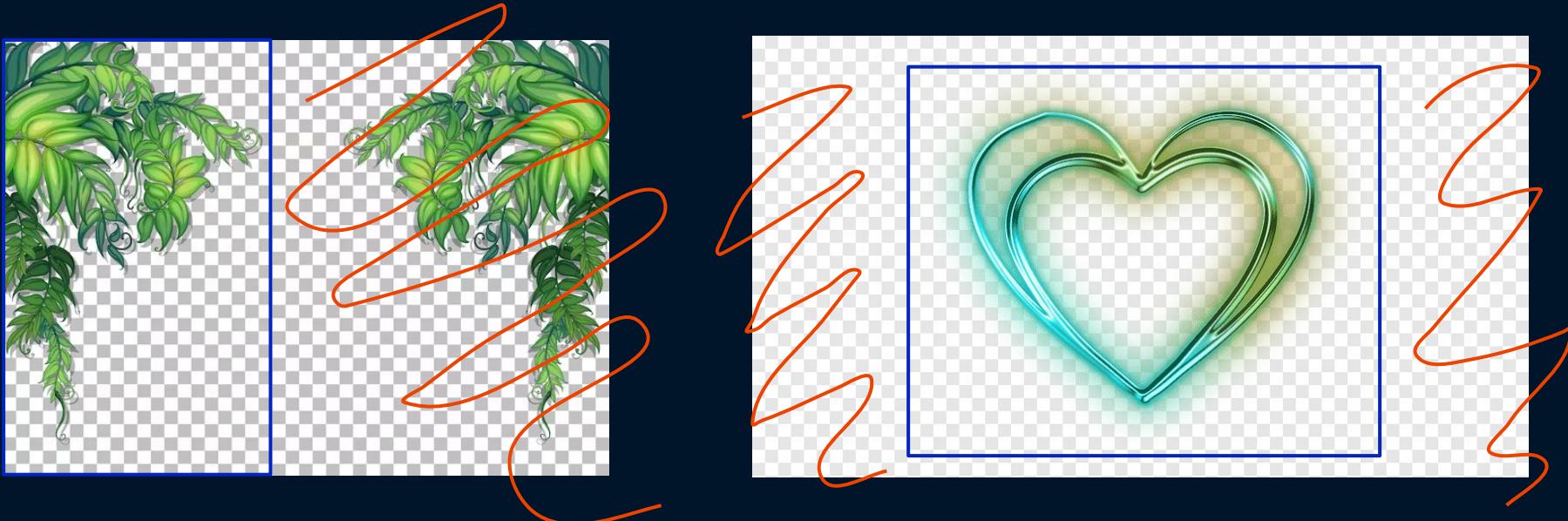
Decomposition



Optimization methods

Crop to exact size

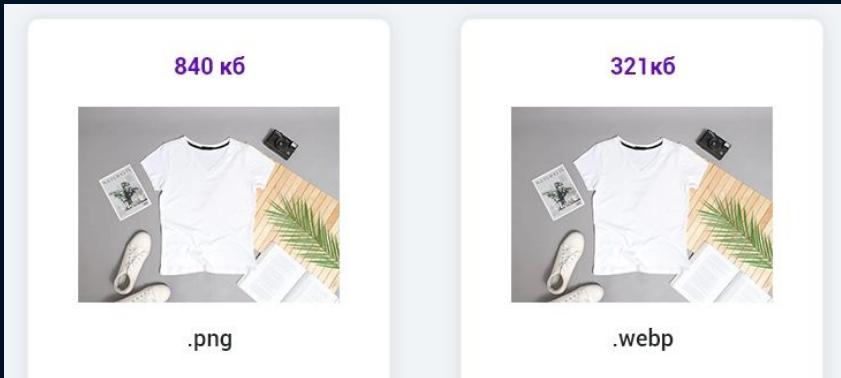
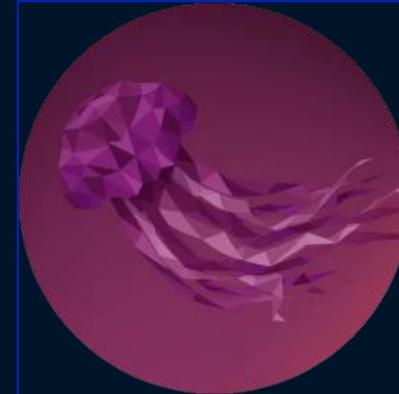
No empty areas or unnecessary transparency



Optimization methods

Change format

- Choose a more modern format
- Convert raster to vector
- Remove unnecessary transparency



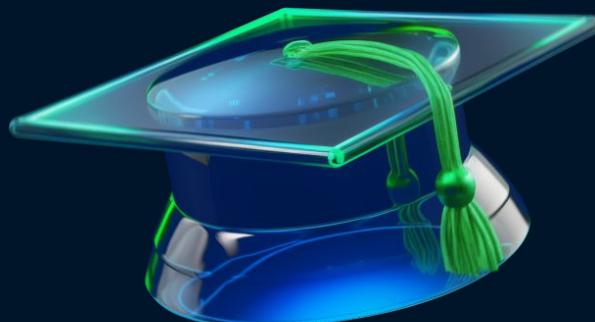
Optimization Methods

- Use CSS Styling Instead of Image Merging:
 - Borders
 - Gradients
 - Filters
 - Transformations (mirror, rotate, tilt)
- Using patterns



Summary

1. Difference between image formats
2. Difference between raster and vector
3. Choosing the right image format
4. Selecting the insertion method for images
5. How and why to optimize images



Quality Criteria for HTML Course

- ❤️ Mandatory for passing the course
- 💛 Required for the highest grade
- 💚 Optional

- ❤️ Correct Image Format
- ❤️ Extract images without style filters from the design.
- ❤️ Correct Image Insertion method
- ❤️ Specify dimensions for all images
- ❤️ Provide alt text for all images.
- 💛 Optimize images for browser rendering.
- 💛 Elements with background images should have a matching background color.
- 💚 Favicon icons display correctly.

Homework

1. Export all images from the Figma template in suitable formats: `webp`, `svg`
2. Insert images into your markup, using the appropriate method of insertion: `img`, `background-image` or `svg`.
3. Add `alt` text in `` tags.
4. Add basic sizes: `width` and `height`, in `` tags.
5. Add `background-color` property for selectors with `background-image`.





QUESTIONS?



Please fill out the feedback form
It's very important for us



THANK YOU!
Have a good evening!

Raster

Images

photos	jpeg, webp, avif
images with transparency	png, webp, avif
images with gradient	webp, avif
large images	webp, avif

