BRO Academy

Sprites

HTML course: Lesson 14 part 2

Lesson Plan

1

The Purpose
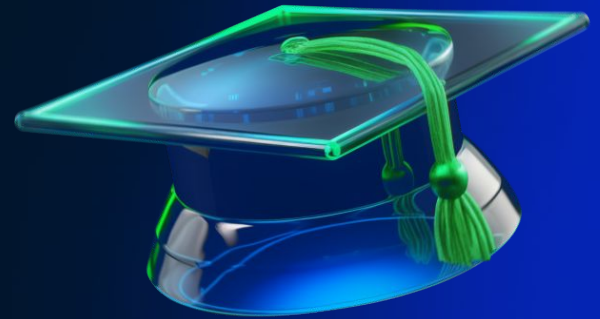
2

Image inlining

3

Raster sprites

4

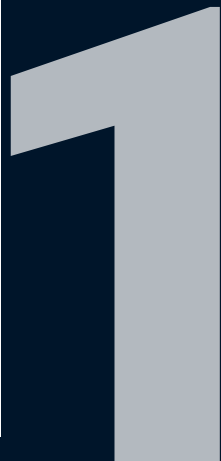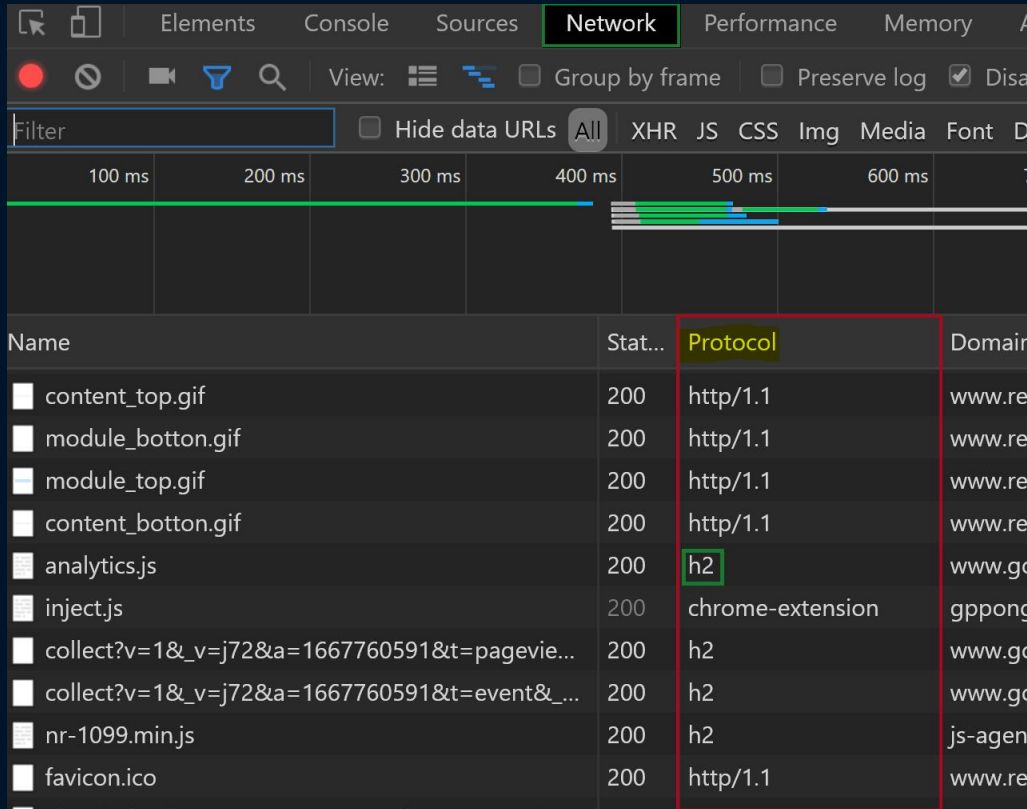Vector sprites

5

Styling

The Purpose

1st problem: too long loading

Previously, we couldn't wait for images to load
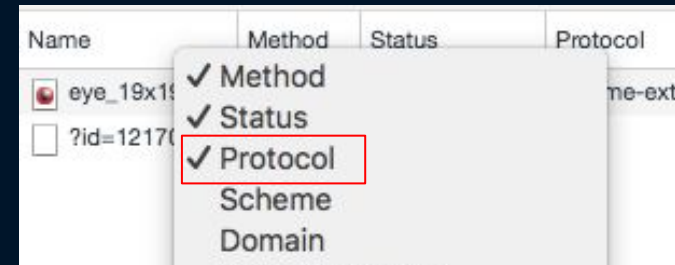
How to check used HTTP protocol version

in DevTools	Real name	Introduced
http/1.1	HTTP/1.1	1997
h2	HTTP/2	2015
h3	HTTP/3	2022

HTTP 1.1 transfer protocol restrictions
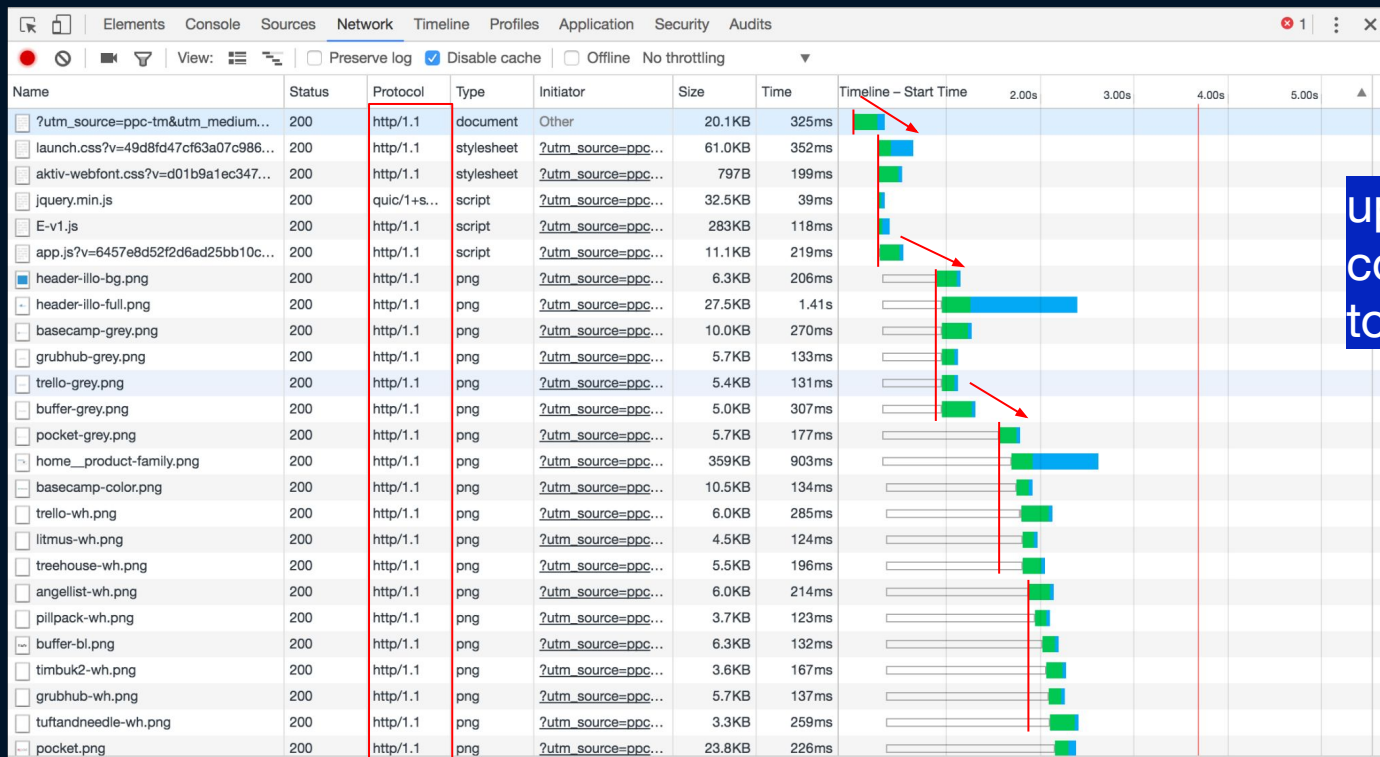
up to ~6 parallel connections to a domain

Image inlining

Data URI

```
<img src="data:[<media-type>][;base64],<data>" alt="logo">
```

base64

```
<img src="data:[<media-type>][;base64],<data>" alt="logo">
```

```
<img src="data:img/png;base64,iVBORw0KGgoAAAANSUhEUgAACAAA
AAgCAMAAABEpIrGAAABO1BMVEU/NFy5BOguxz1eFmMqJnD7ctI1mK3/gf6e
wKFQpwQIHWi1BUsiVTYrHr/j88JngEGdlzVo33v+JvN0xbq1+b9nQ10ofsI
dwN89u6keUO8FD8M65/ZVS0hhpqwKse7Exw0AAABW0lEQVQ4jX2TC1OCQBS
Ft5qyUppqnF72JHN6OVaMxwiLAkuNMs3U1B5WZv3/X9AuAhKLHGbg7pxvzt
0LLCFU0bW6Vte8UseIrQx8FXOAI00XAEPX49ZF3XK8NukApJK6AqZSqbylO
oxm/pW4NQJcOoswcPvfNoGGXStbwLzHJyVXglwDdgOARA5QZS9wCCStMloG
Nr0+A6yE+xmgWOGAczshewyIYc53WkS69CV1eZ8lPLBni/pqZBigTBeof8p
NwPQIXDdm2Sc62=" alt="logo">
```
iVBORw0KGgoAAAANSUhEUgAACAAAAAgCAMAAABEpIrGAAABO1BMVEU/NFy5BOguxz1eFmMqJnD7ctI1mK3/gf6ewKFQpwQIHWi1BUsiVTYrHr/j88JngEGdlzVo33v+JvN0xbq1+b9nQ10ofsIdwN89u6keUO8FD8M65/ZVS0hhpqwKse7Exw0AAABW0lEQVQ4jX2TC1OCQBSFt5qyUppqnF72JHN6OVaMxwiLAkuNMs3U1B5WZv3/X9AuAhKLHGbg7pxvzt0LLCFU0bW6Vte8UseIrQx8FXOAI00XAEPX49ZF3XK8NukApJK6AqZSqbylOoxm/pW4NQJcOoswcPvfNoGGXStbwLzHJyVXglwDdgOARA5QZS9wCCStMloGNr0+A6yE+xmgWOGAczshewyIYc53WkS69CV1eZ8lPLBni/pqZBigTBeof8pNwPQIXDdm2Sc62=

URL Encoded (percent encoded)

```
<img src="data:[<media-type>][;base64],<data>" alt="logo">


<img src="data:image/svg+xml;utf8,<svg width='10'...
</svg>" alt="logo">
```

```
background-image: url("data:image/svg+xml;utf8,<svg
width='10'... </svg>");
```

list-style-image
border-image
content

important to replace
all these symbols:
%#()<>?[\]^`{|}

URL encoder: yoksel.github.io/url-encoder

from	to
:	%3A
/	%2F
' '	%20 or +

1st problem was reduced ✅

Inline images shown faster

1

2nd problem: this postpones everything else on a page

if you have a lot of images, all resources will be shown later

so, it's bad idea to put all the images inline into HTML

2

Raster sprites

what if there will be only one image?
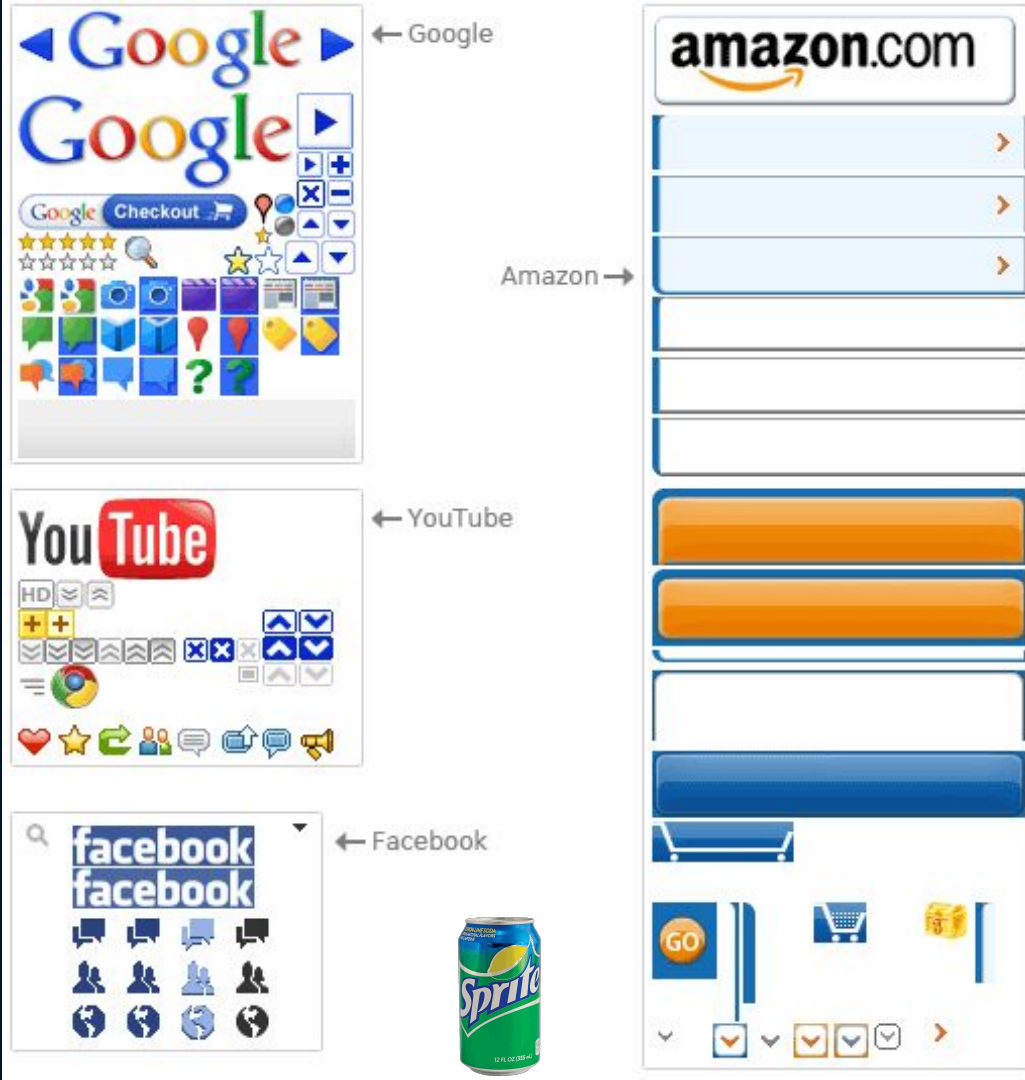
just glue them together

2

Sprite

use sprites to join all images network requests

Generator: spritecow.com

when to use:
- to show all icons at the same time

Sprite with img + object-position

```
<img class="sprite mail" src="sprite.webp" alt="">
```
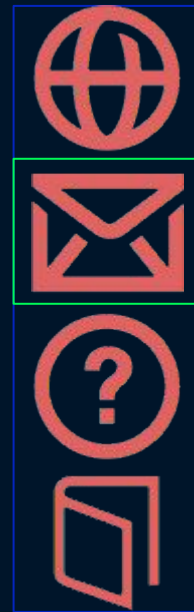
```css
.sprite {
    width: 75px; height: 75px;
    object-fit: none;
}

.mail { object-position: 0 -75px }
```

Same with background

```
<img class="sprite world" src="sprite.webp" alt="">


.sprite {
    width: 75px; height: 75px;
    background-image: url(sprite.webp);
}

.mail { background-position: 0 -75px }
```

example →

1st and 2nd problems were reduced ✅

just glued all the images and inlined into HTML

2

3rd problem: gifs

when you need to show animated images, GIF is bad word

3

Vertical, horizontal sprites

can be done with CSS animations (steps)
this technique also known as sequence

when to use:
- for animations
  with less than 60 frames,
  better up to 30

3st problem was solved ✅

what if you need to show animated images?


3

HTTP 1.1 vs **HTTP 2** difference

| | Elements | Console | Sources | Network | Timeline | Profiles | Application | Security | Audits | | | | ⊗ 1 | ⋮ | ✕ |

● ⊘ ■ ▼ View: ☰ ☴ □ Preserve log ☑ Disable cache □ Offline No throttling ▼

Name	Status	Protocol	Type	Initiator	Size	Time	Timeline – Start Time	1.00s	1.▲
delightfuldownloads.com	200	http/1.1	document	Other	9.8KB	46ms			
edd.min.css	200	h2	stylesheet	(index):39	9.6KB	372ms			
style.min.css	200	h2	stylesheet	(index):40	14.0KB	314ms			
style.min.css	200	h2	stylesheet	(index):41	2.4KB	245ms			
css?family=Montserrat%3A400%2C...	200	quic/1+s...	stylesheet	(index):42	1.7KB	66ms			
font-awesome.min.css	200	h2	stylesheet	(index):43	5.7KB	291ms			
lightGallery.min.css	200	h2	stylesheet	(index):44	5.5KB	223ms			
style.min.css	200	h2	stylesheet	(index):45	644B	228ms			
jquery.js	200	h2	script	(index):59	33.4KB	404ms			
jquery-migrate.min.js	200	h2	script	(index):60	4.3KB	218ms			
edd-ajax.min.js	200	h2	script	(index):66	2.8KB	221ms			
searchwp-live-search.min.js	200	h2	script	(index):73	3.9KB	207ms			
hoverIntent.min.js	200	h2	script	(index):360	855B	217ms			
checkout.min.js	200	h2	script	(index):366	1.9KB	217ms			
imagesloaded.min.js	200	h2	script	(index):367	2.8KB	206ms			
masonry.min.js	200	h2	script	(index):368	8.7KB	262ms			
jquery.matchHeight.min.js	200	h2	script	(index):369	1.6KB	977ms			
responsiveslides.min.js	200	h2	script	(index):370	2.1KB	166ms			
jquery.touchSwipe.min.js	200	h2	script	(index):371	5.0KB	178ms			
fastclick.min.js	200	h2	script	(index):372	3.2KB	149ms			
lightGallery.min.js	200	h2	script	(index):373	1.3KB	149ms			
jquery.fitvids.min.js	200	h2	script	(index):374	1.4KB	165ms			
edd.min.js	200	h2	script	(index):380	2.0KB	177ms			
wp-embed.min.js	200	h2	script	(index):381	1.1KB	218ms			
forms-api.min.js	200	h2	script	(index):387	5.2KB	1.11s			

All the problems were completely solved ✅

4th problem: sprite consists of same icons

we gathered a lot of icons in our sprites,
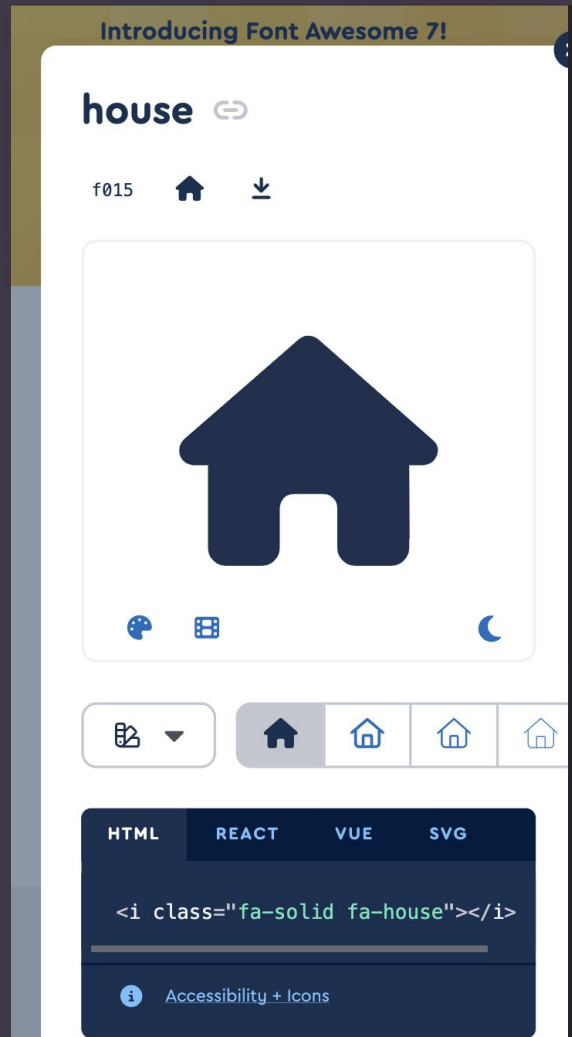and some icons were repeated with just changing some color
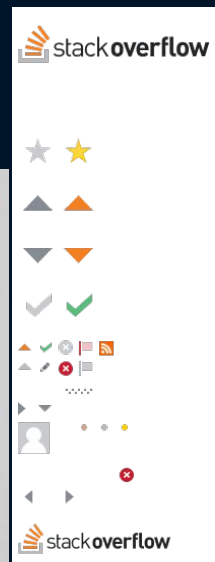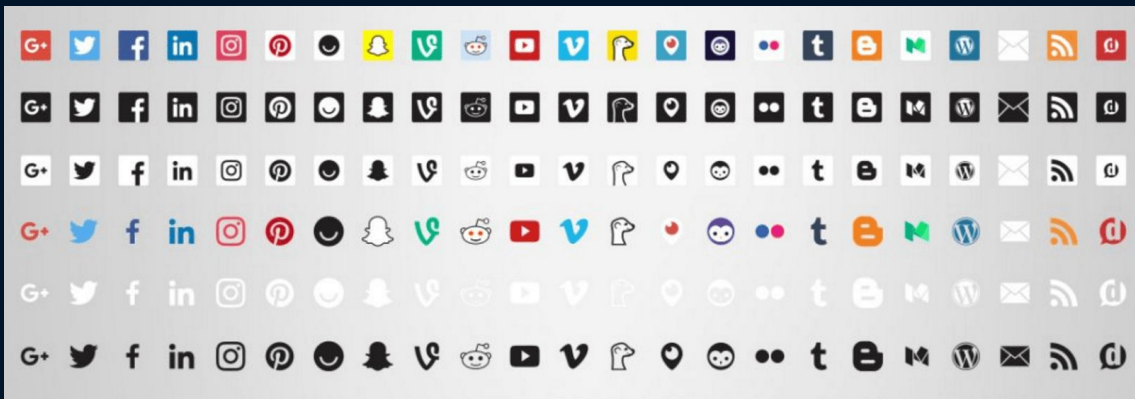
Icon fonts

Huge font file
with all the icons
suitable for all projects

❌ never use


```
element.style {
    line-height: 1;
    color: □ rgb(30, 48, 80);
}

:is(i[class*="fa-"], .svg-          main.css:1671
inline--fa) {
    --spacing-inline: var(--spacing-6xs);
}

.fa-solid, .fas {                   all.css:9
    font-weight: 900;
}

.fa-home, .fa-home-alt, .fa-home-   all.css:8
lg-alt, .fa-house {
    --fa: "\f015";
    --fa--fa: "\f015\f015";
}

.fa-classic, .fa-light, .fa-        all.css:6
regular, .fa-solid, .fa-thin, .fal,
.far, .fas, .fat {
    font-family: "Font Awesome 6 Pro";
}

.fa, .fa-brands, .fa-classic, .fa-  all.css:6
duotone, .fa-light, .fa-regular,
.fa-sharp, .fa-sharp-duotone, .fa-solid, .fa-
thin, .fab, .fad, .fadl, .fadr, .fadt, .fal,
.far, .fas, .fasdl, .fasdr, .fasds, .fasdt,
.fasl, .fasr, .fass, .fast, .fat {
    -moz-osx-font-smoothing: grayscale;
    -webkit-font-smoothing: antialiased;
    display: var(--fa-display, inline-block);
    font-style: normal;
    font-variant: ▶ normal;
    line-height: 1;
    text-rendering: auto;
}

i, .i, em, .em {                    main.css:2533
    --font-style: italic;
    font-style: italic;
    font-style: var( font-style);
}

*, *::before, *::after {            main.css:139
    box-sizing: border-box;
}

i {                                 user agent stylesheet
```

Introducing Font Awesome 7!

house 🔗

f015 🏠 ⬇

🏠

🎨 🎞 🌙

HTML   REACT   VUE   SVG

`<i class="fa-solid fa-house"></i>`

ℹ Accessibility + Icons

CSS Filters

- works unpredictable in safari

- don't transition filters
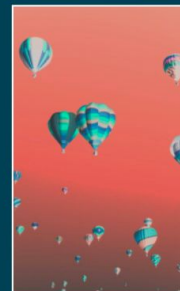
Break time

Vector sprites

Old way: defs + use

```svg
<svg viewBox="0 0 100 100" fill="none" xmlns="http://www.w3.org/2000/svg">
    <defs>
        <g id="heart">
            // …
        </g>
        <g id="star">
            // …
        </g>
    </defs>
</svg>

<svg class="icon">
    <use xlink:href="./sprite.svg#heart"></use>
</svg>

<svg class="icon">
    <use xlink:href="#heart"></use>
</svg>
```

definitions are hidden

icons are with id

and used by id

Old way: stacks with :target

```svg
<svg viewBox="0 0 100 100" fill="none" xmlns="http://www.w3.org/2000/svg">
    <defs>
        <style><![CDATA[
            .sprite { display: none; }
            .sprite:target { display: inline; }
        ]]></style>
    </defs>
    <g class="sprite" id="heart">...</g>
    <g class="sprite" id="star">...</g>
</svg>

<svg class="icon">
    <use xlink:href="./sprite.svg#heart"></use>
</svg>
```
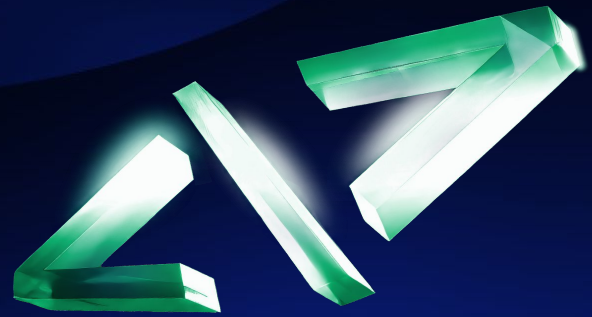
New way: symbols + use

✨ best approach

```svg
<svg viewBox="0 0 100 100" fill="none" xmlns="http://www.w3.org/2000/svg">
    <symbol id="heart">
        // …
    </symbol>
    <symbol id="star">
        // …
    </symbol>
</svg>

<svg class="icon">
    <use xlink:href="./sprite.svg#heart"></use>
</svg>


<svg class="icon">
    <use href="./sprite.svg#heart"></use>
</svg>
```

Styling

Change icon colors on hover

```css
.icon {
    fill: blue;
    transition: fill 0.3s, stroke 0.3s;
}


.icon:hover {
    fill: red;
}
```

color: currentColor

replace colors inside of SVGs in sprite folder to currentColor

```
<svg viewBox="0 0 512 512" class="logo__icon">
    <path fill="currentColor" d="m48 300 180 169a41 41
0 0 0 56 0l180-169c31-28 48-68 48-109v-6A143 143 0 0 0
268 84l-12 12-12-12A143 143 0 0 0 0 185v6c0 41 17 81 48
109z"/>
</svg>
```

Custom properties

replace colors inside of SVGs in sprite folder to custom properties
with fallback color

```
<svg viewBox="0 0 512 512" class="logo__icon">
    <path fill="var(--icon-color, blue)" d="m48 300 180
169a41 41 0 0 0 56 0l180-169c31-28 48-68 48-109v-6A143 143
0 0 0 268 84l-12 12-12-12A143 143 0 0 0 0 185v6c0 41 17 81
48 109z"/>
</svg>
```
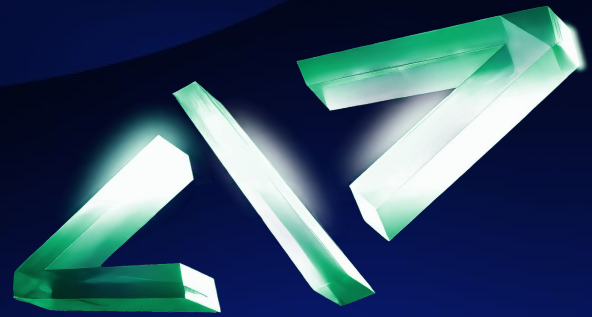
Custom properties + currentColor

replace colors inside of SVGs in sprite folder to custom properties
with fallback color as a currentColor

```html
<svg viewBox="0 0 512 512" class="logo__icon">
    <path fill="var(--icon-color, currentColor)" d="m48
300 180 169a41 41 0 0 0 56 0l180-169c31-28 48-68
48-109v-6A143 143 0 0 0 268 84l-12 12-12-12A143 143 0 0 0
0 185v6c0 41 17 81 48 109z"/>
</svg>
```

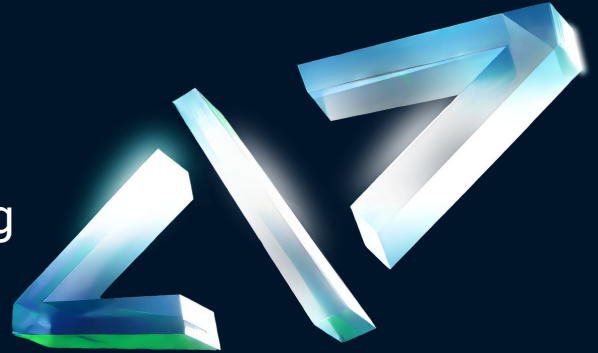# All the problems were completely solved ✅

Build

Our build

let's try to put some icons into

sprite folder

Homework

1. Put all sprite vector images into the `src/images/sprite` folder.

2. Use them on pages from sprite via svg use tag.

3. Implement color changes on hover and focus using only CSS for icons that need interactive effects.

Quality Criteria for HTML Course

❤️ Mandatory for passing the course

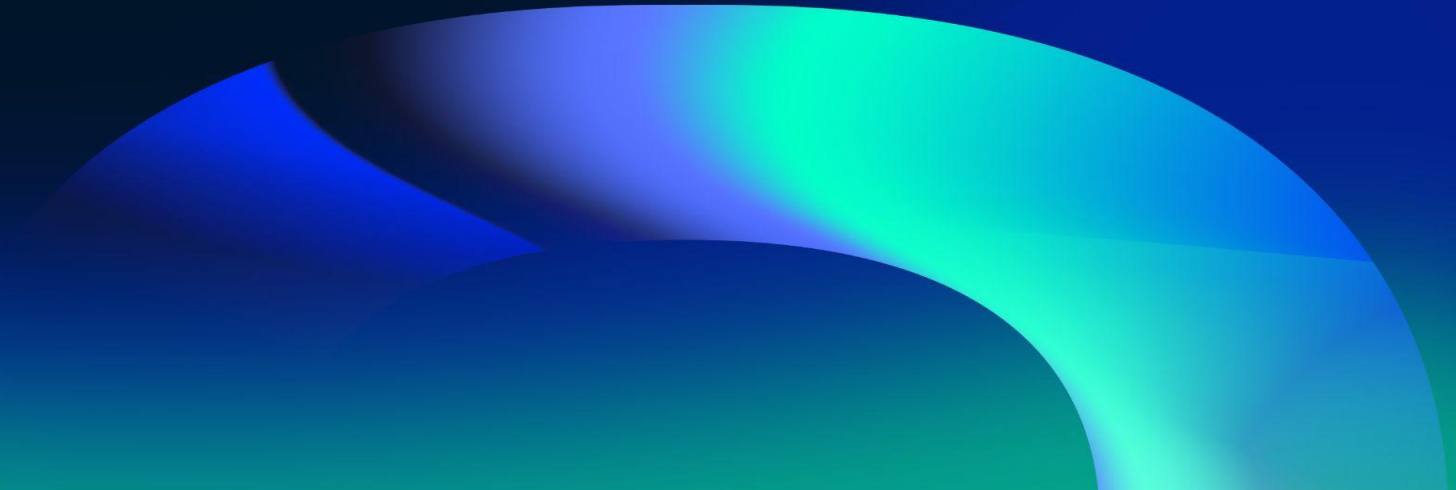💛 Required for the highest grade

💚 Optional

❤️ Use SVG sprite with symbols

Please fill out the feedback form

It's very important for us

THANK YOU!
Have a good evening!