Webudvikling Frontend

CSS, fonts, responsiveness & Design Principles

Webudvikling

KØBENHAVNS ERHVERVSAKADEMI

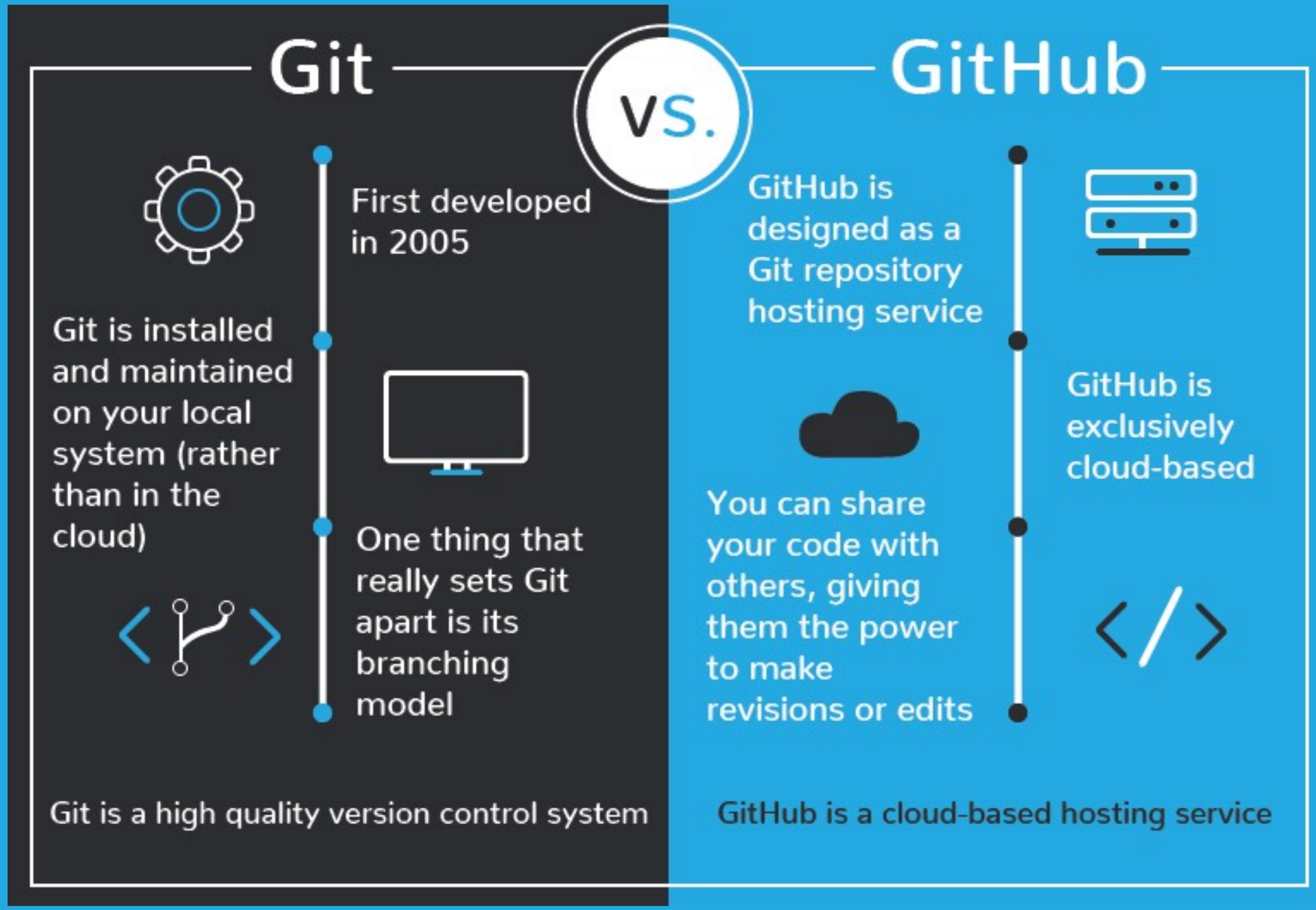
Dagens Formål

- Opsamling - hands-on med moderne semantisk HTML
- Viden om performance og Web Dev Tools,
- Intro og hands-on med CSS, fonts og responsivt design
- Erfaring med værktøjer
- Og hvad med designprincipper?

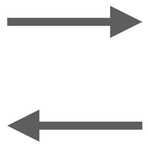
Agenda

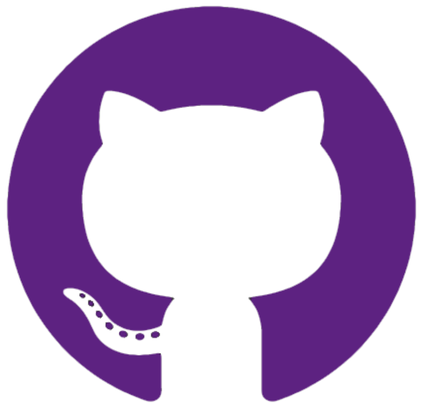
- Opsamling
- HTML & HTML Semantics
- Performance & Dev Tool
- CSS
- Fonts
- Responsive Web Design

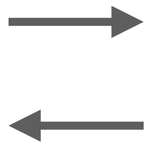

Git, GitHub & GitHub Desktop
Version Control, Collaboration & how to manage your code projects


Git
vs.
GitHub
First developed in 2005
Git is installed and maintained on your local system (rather than in the cloud)
One thing that really sets Git apart is its branching model
Git is a high quality version control system
GitHub is designed as a Git repository hosting service
GitHub is exclusively cloud-based
You can share your code with others, giving them the power to make revisions or edits
GitHub is a cloud-based hosting service

Git







WEB PROJECT

GITHUB DESKTOP
APP / GUI

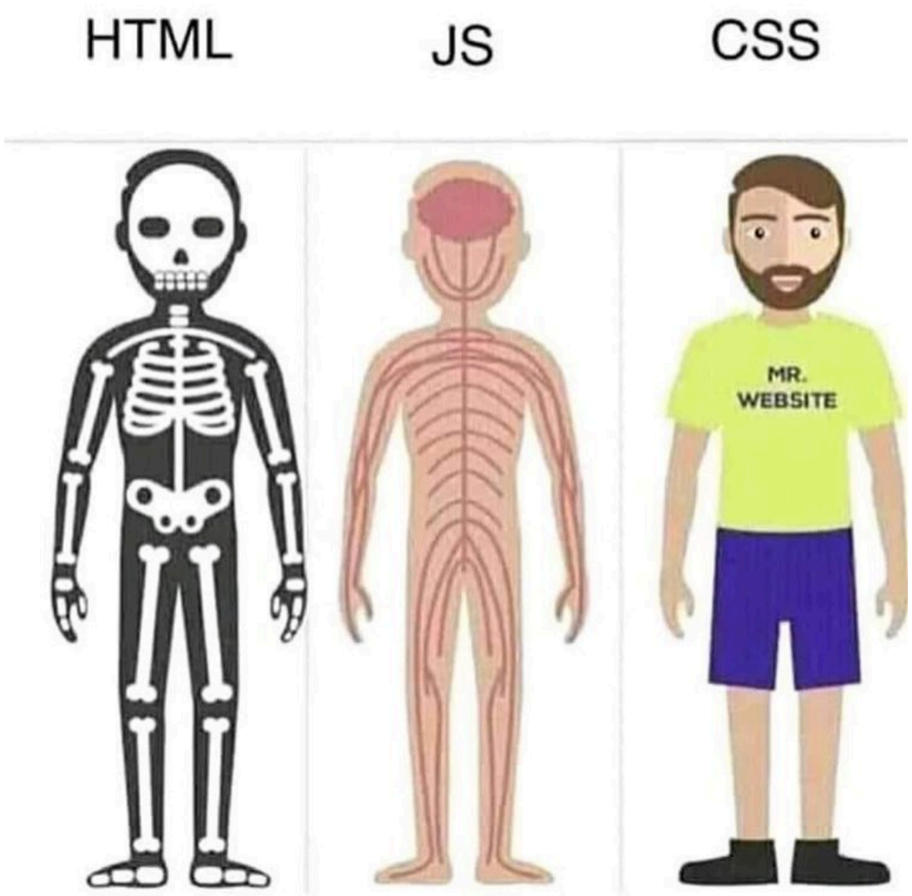
GITHUB
SERVER WITH REPOSITORIES

Frontend stack

STRUCTURE
CONTENT

FUNCTIONALITY
BEHAVIOR

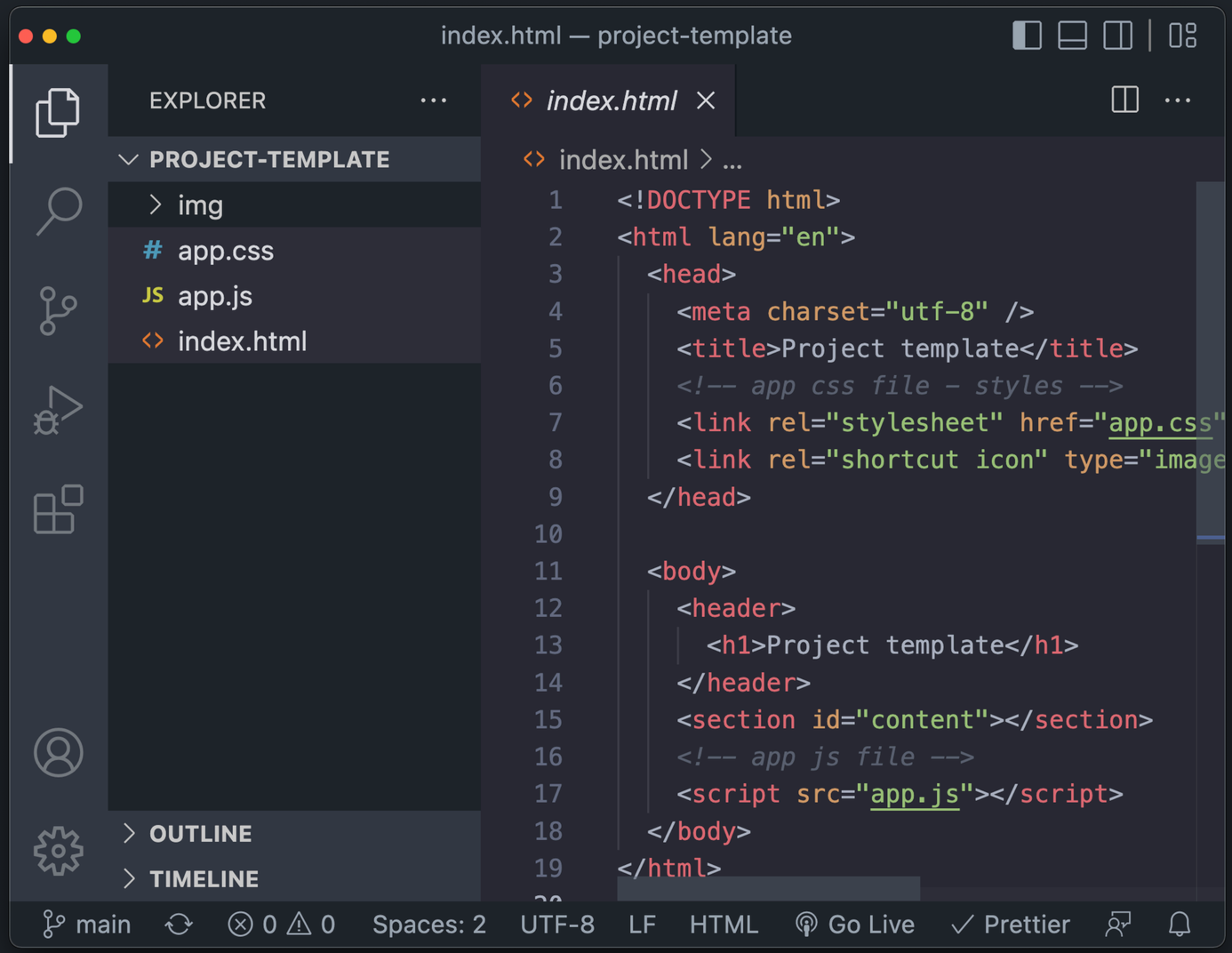
LAYOUT/STYLING
PRESENTATION



https://www.instagram.com/p/Bw90x2Ojo4g/

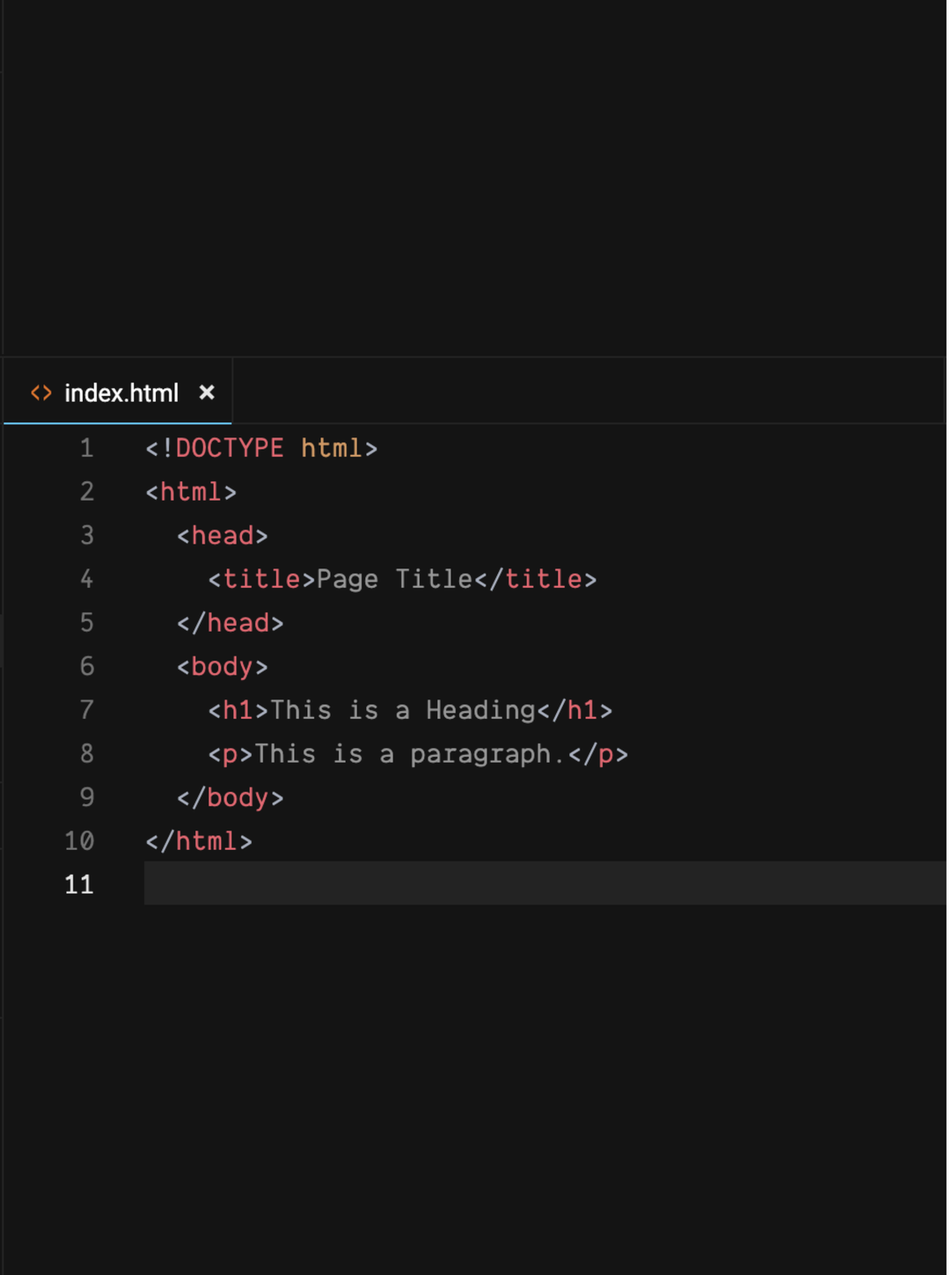
Frontend Project Structure

- Project structure
  - HTML
  - CSS
  - JavaScript
- Keep a good structure & strive for consistency
- Separation of concerns



HTML, CSS & JavaScript

You can literally build anything with it!



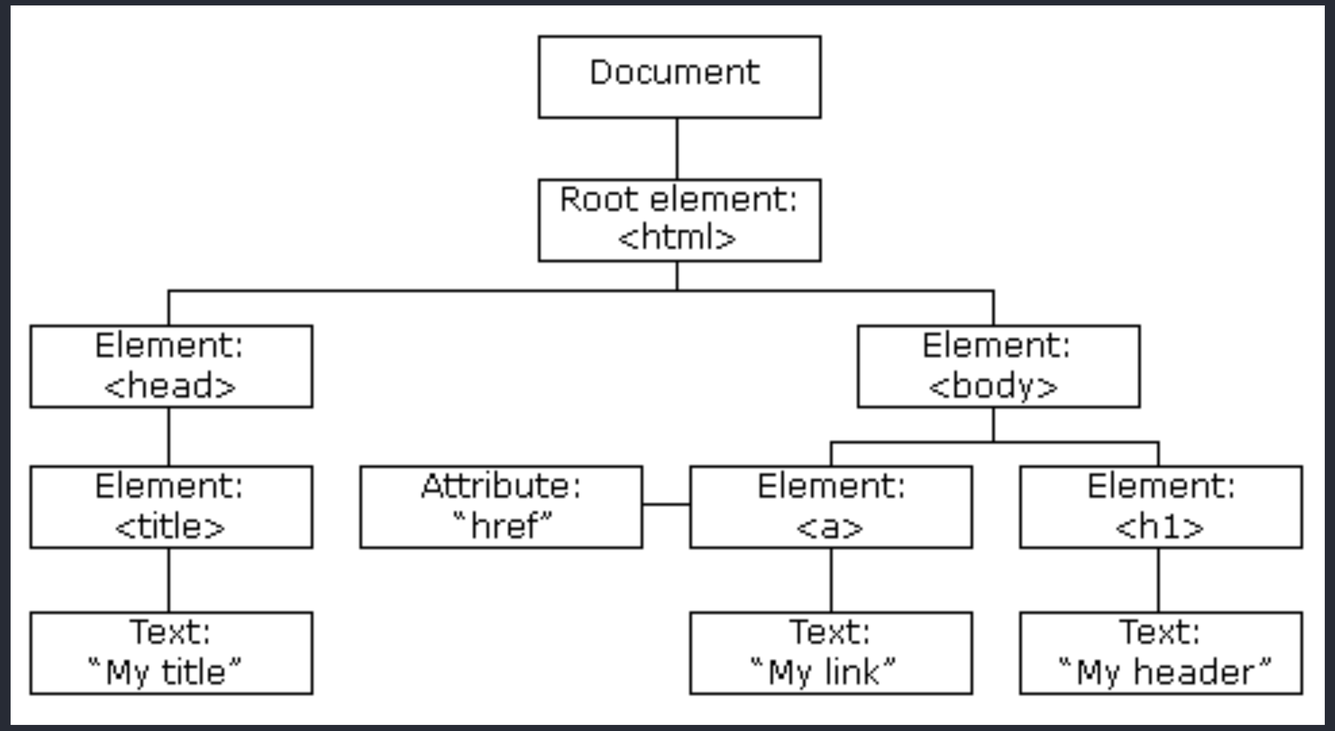
What is HTML?

- Hyper Text Markup Language
- Standard markup language for creating Web pages
- Describes the structure of a Web page
- Consists of a series of elements (tags)
- HTML elements tell the browser how to display the content

https://www.w3schools.com/html/html_intro.asp

The HTML DOM (Document Object Model)

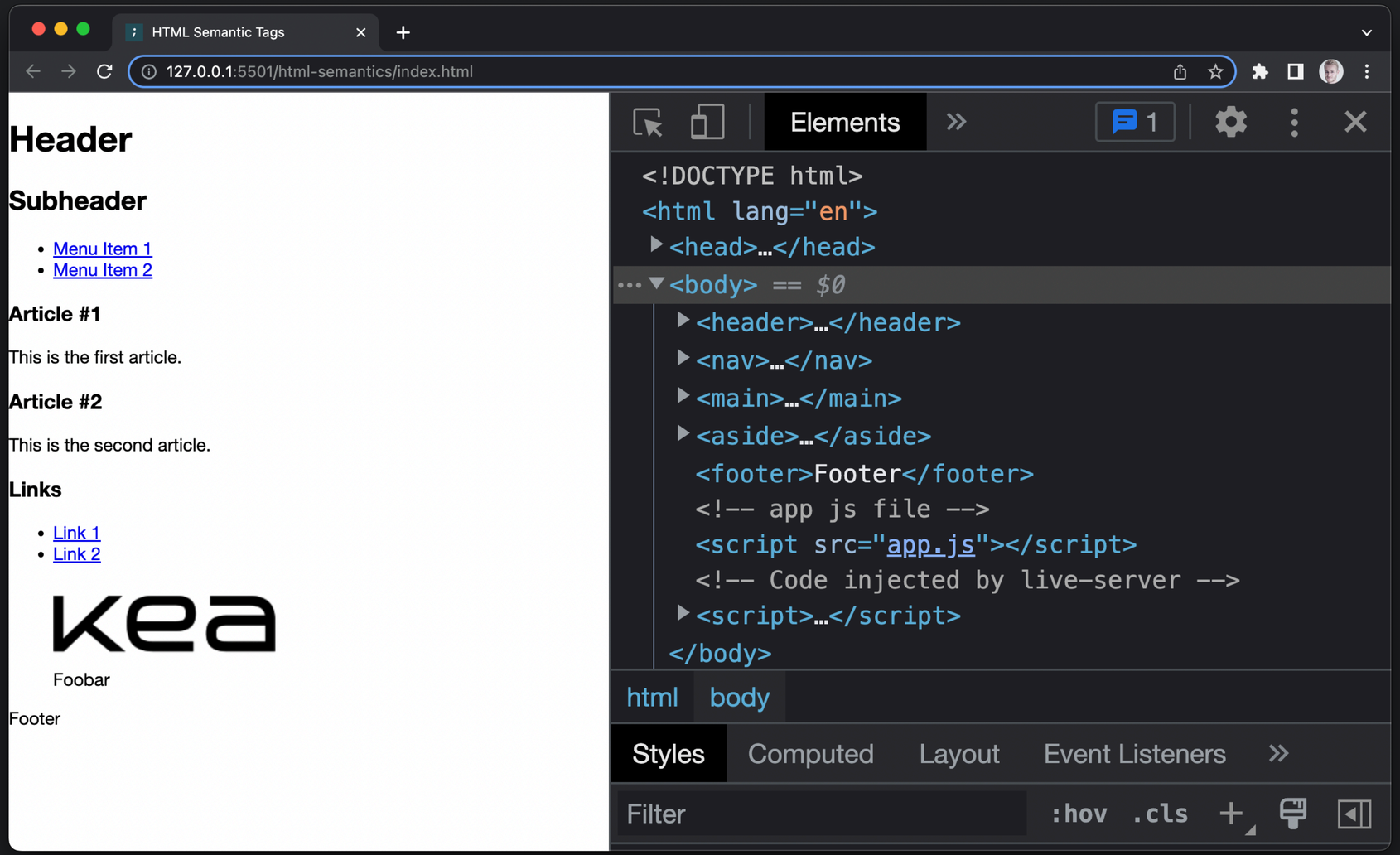
A model as a tree of Objects



- Object Model for HTML:
  - HTML elements as objects
  - Properties for all HTML elements
  - Methods for all HTML elements
  - Events for all HTML elements

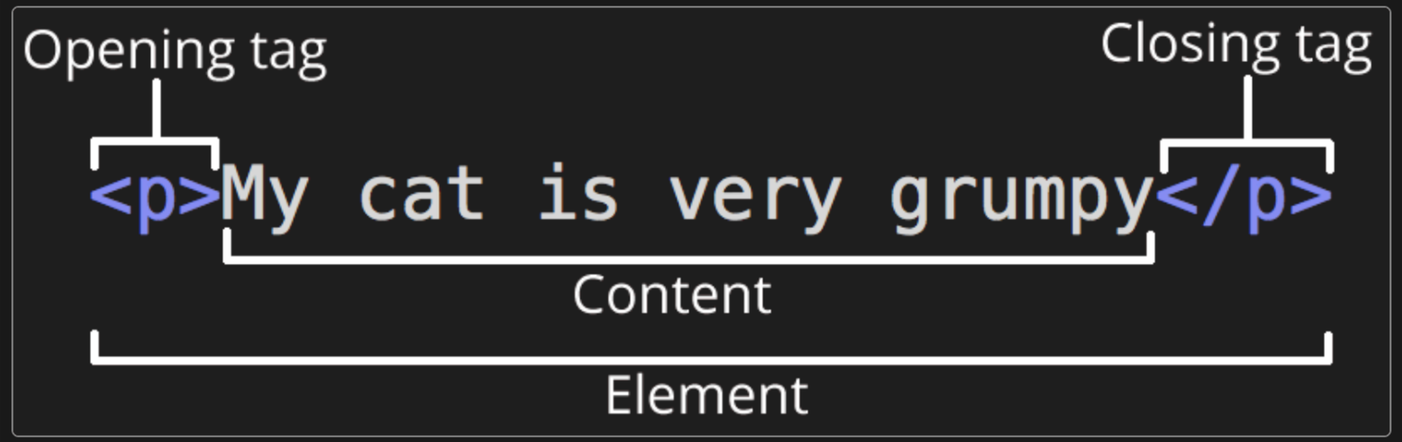
https://www.w3schools.com/whatis/whatis_htmldom.asp

Dev Tool: Elements



Anatomy of an HTML element

Element: opening and closing tag + content



https://developer.mozilla.org/en-US/docs/Learn/HTML/Introduction_to_HTML/Getting_started#anatomy_of_an_html_element

Void elements

Element: single tag, usually to insert or embed something in a HTML

```
<img src="https://kompetence.kea.dk/images/resources/logo-main-black-single.png" alt="KEA logo" />
```

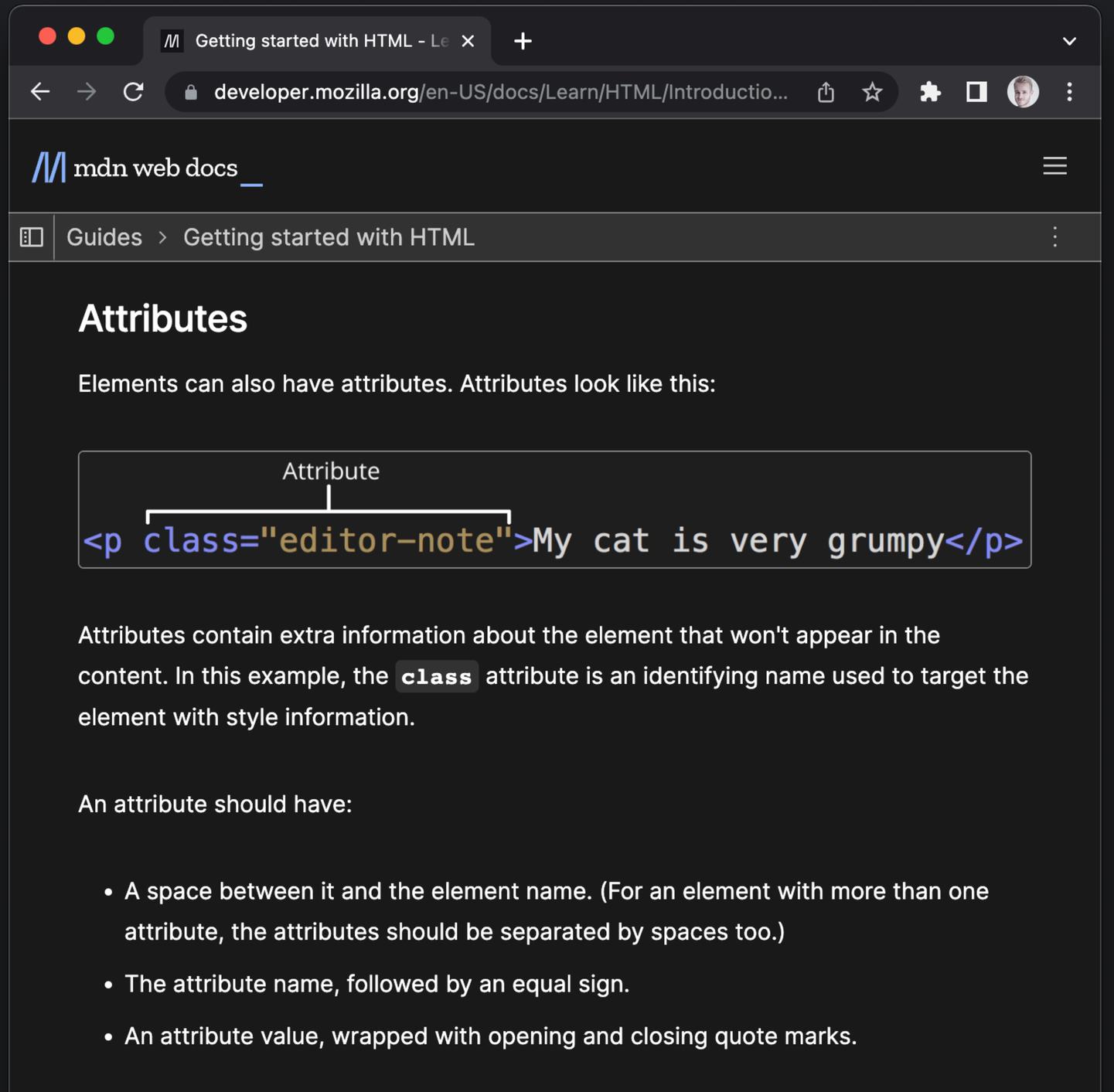
https://developer.mozilla.org/en-US/docs/Learn/HTML/Introduction_to_HTML/Getting_started#void_elements

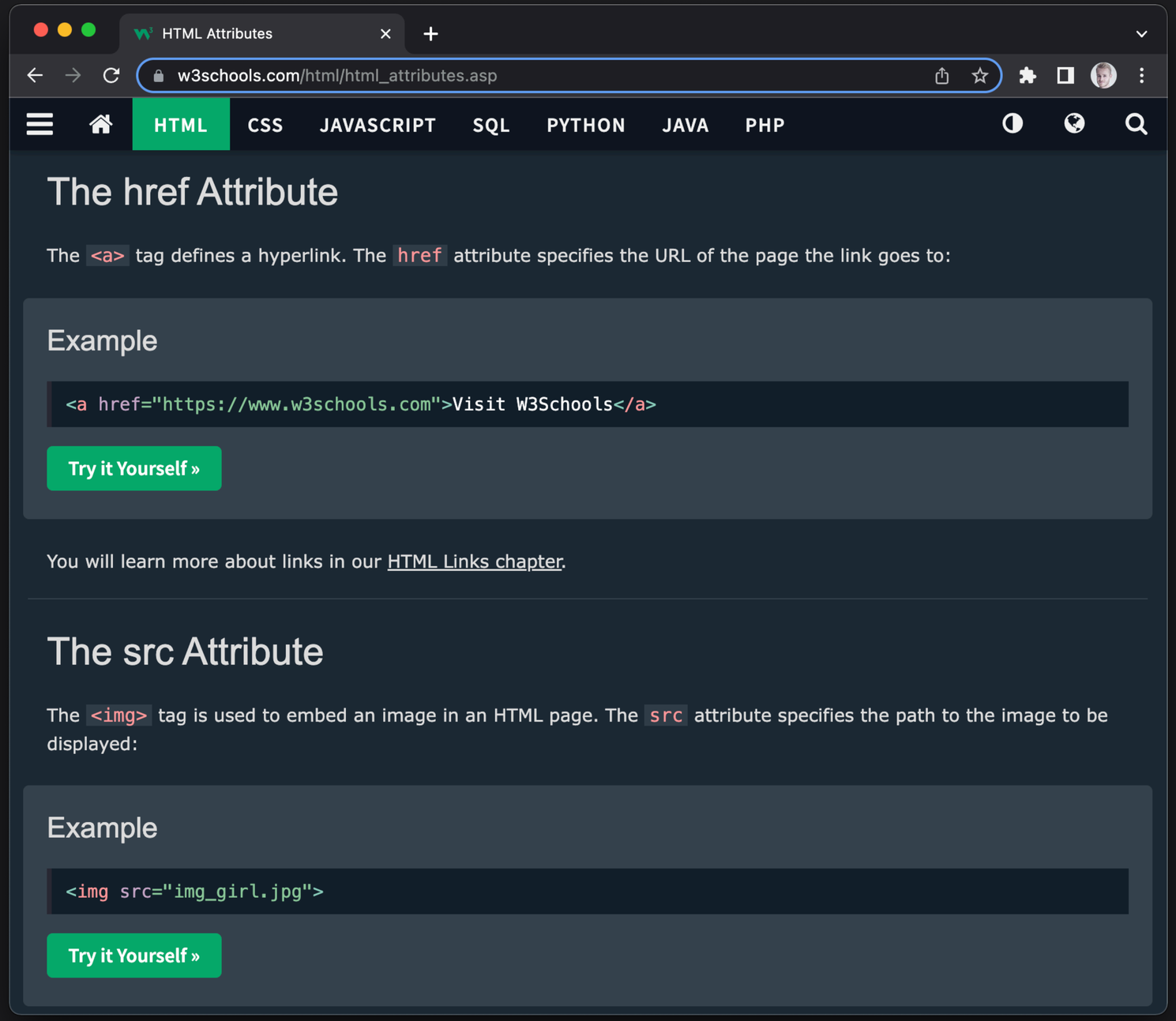
Attributes

- Provides additional information and all element can have them.
- Are specified in the start tag
- name/value pairs: `name="value"`

Attributes - mdn

HTML Attributes - W3Schools





https://www.w3schools.com/html/html_attributes.asp

index.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Page Title</title>
  </head>
  <body>
    <h1>This is a Heading</h1>
    <p>This is a paragraph.</p>
  </body>
</html>

```

Browser Tests

https://9nypo.csb.app/

This is a Heading

This is a paragraph.

Anatomy of an HTML document

index.html

```html
<!DOCTYPE html>
<html>
  <head>
    <title>Page Title</title>
  </head>
  <body>
    <h1>This is a Heading</h1>
    <p>This is a paragraph.</p>
  </body>
</html>
```

The `<!DOCTYPE html>` declaration defines that this document is an HTML5 document

The `<html>` element is the root element of an HTML page

The `<head>` element contains meta information about the HTML page

The `<title>` element specifies a title for the HTML page (which is shown in the browser's title bar or in the page's tab)

The `<body>` element defines the document's body, and is a container for all the visible contents, such as headings, paragraphs, images, hyperlinks, tables, lists, etc.

The `<h1>` element defines a large heading

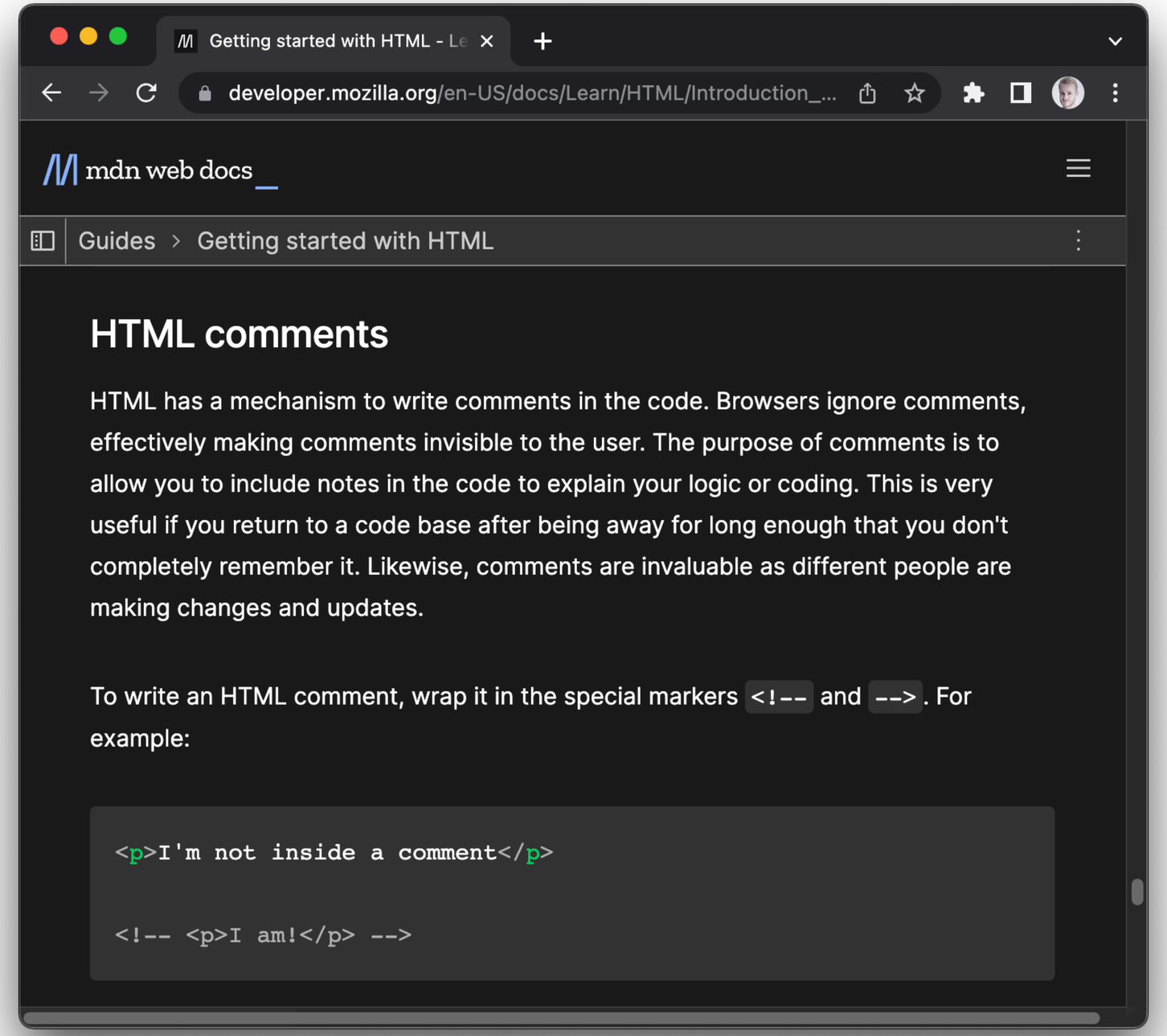
The `<p>` element defines a paragraph

https://www.w3schools.com/html/html_intro.asp

index.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Page Title</title>
  </head>
  <body>
    <h1>This is a Heading</h1>
    <p>This is a paragraph.</p>
  </body>
</html>
```

https://www.w3schools.com/html/html_intro.asp



https://developer.mozilla.org/en-US/docs/Learn/HTML/Introduction_to_HTML/Getting_started#anatomy_of_an_html_document

HTML Quiz & Exercises

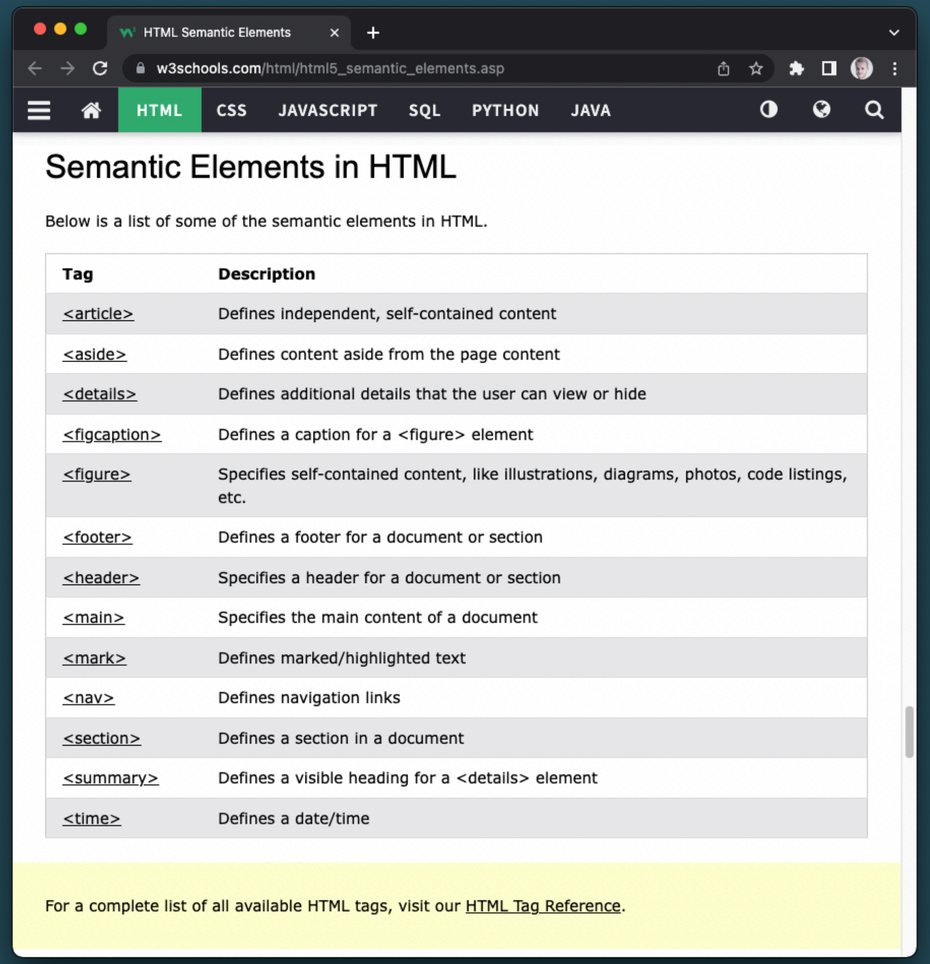
HTML Quiz

HTML Exercises

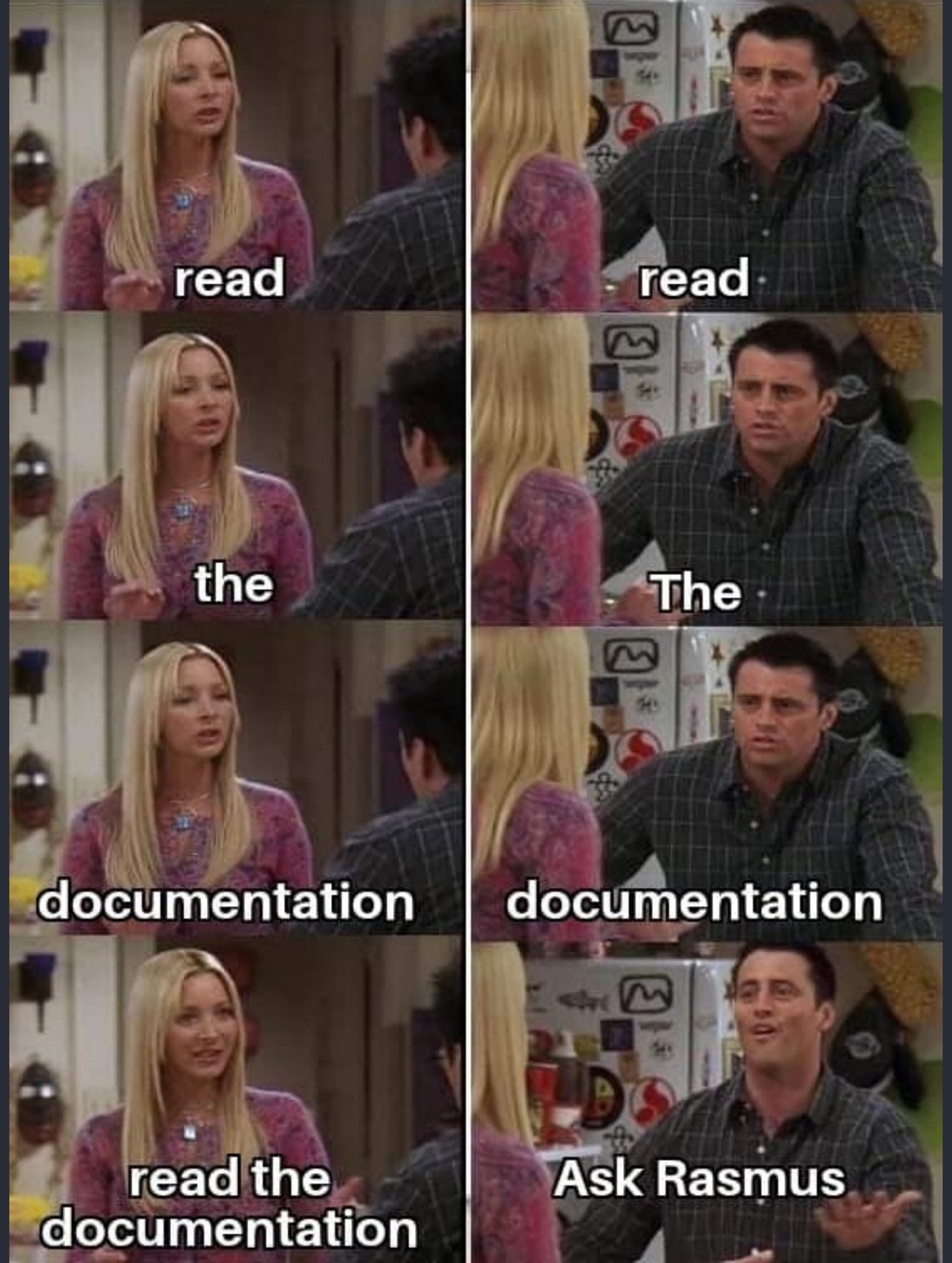
HTML Semantic Elements

Clearly describes the meaning to both the browser and the developer.

Semantic elements = elements with a meaning



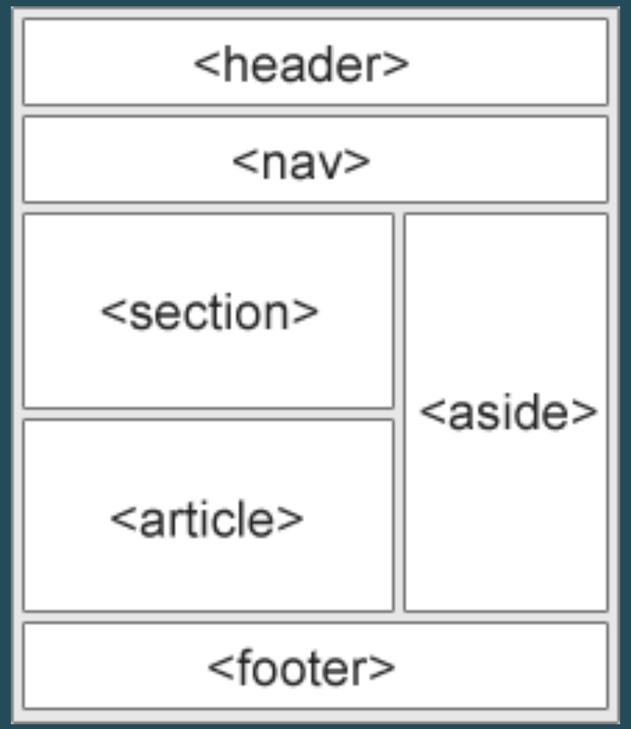
https://www.w3schools.com/html/html5_semantic_elements.asp


read
read
the
The
documentation
documentation
read the documentation
Ask Rasmus

HTML Semantic Elements

Semantic elements that can be used to define different parts of a website

- <article>
- <aside>
- <details>
- <figcaption>
- <figure>
- <footer>
- <header>
- <main>
- <mark>
- <nav>
- <section>
- <summary>
- <time>

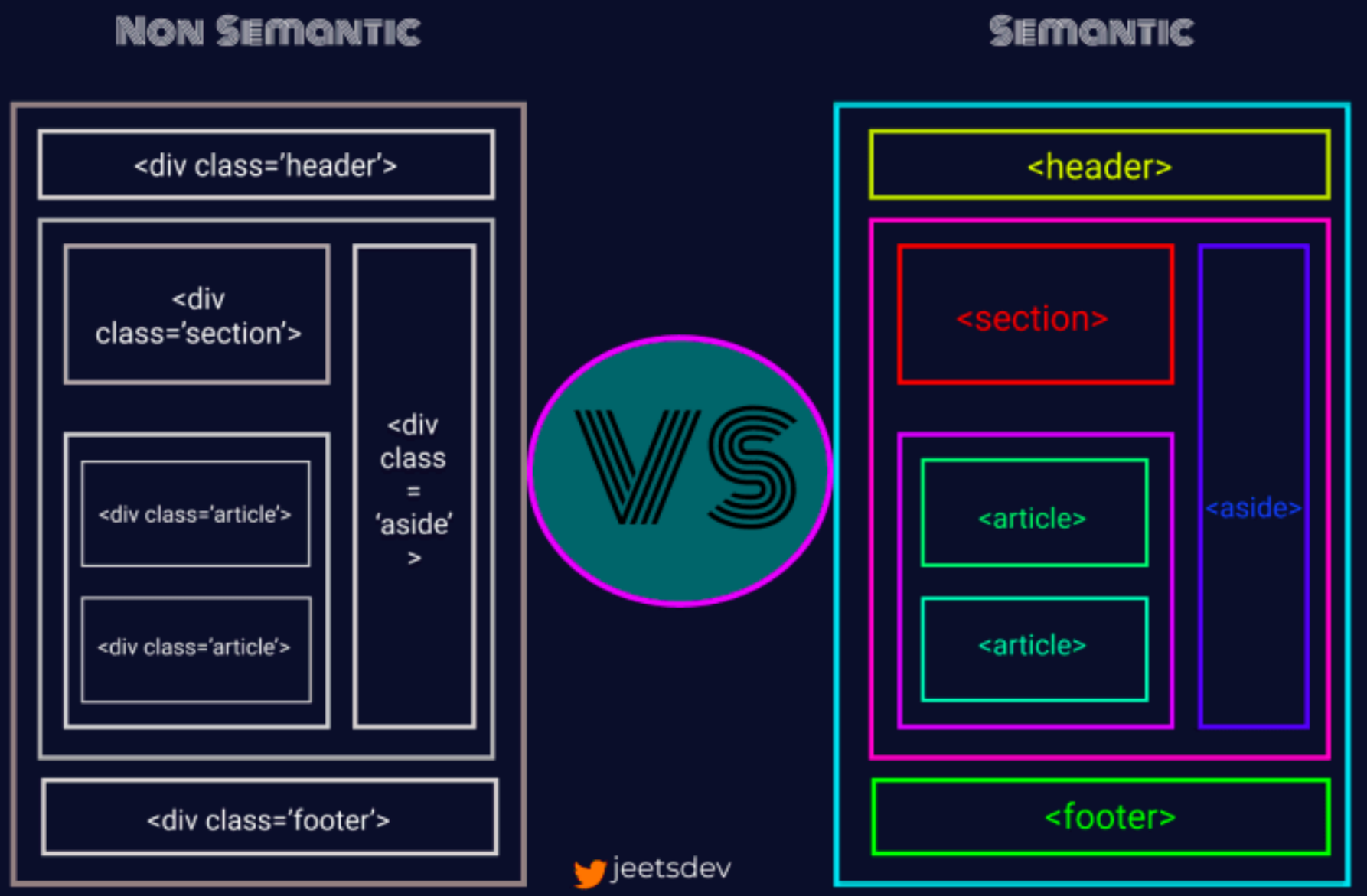


https://www.w3schools.com/html/html5_semantic_elements.asp

Why use semantic elements?

```
<body>
  <div id="header">
    <h1>PROJECT TEMPLATE</h1>
  </div>
  <div class="sections">
    <div class="article">
      <div class="figure">
        <img>
        <div class="figcaption"></div>
      </div>
    </div>
  </div>
  <div id="footer"></div>
  <!-- main js file -->
  <script src="js/main.js"></script>
</body>
```

```
<body>
  <header>
    <h1>PROJECT TEMPLATE</h1>
  </header>
  <section>
    <article>
      <figure>
        <img>
        <figcaption></figcaption>
      </figure>
    </article>
  </section>
  <footer></footer>
  <!-- main js file -->
  <script src="js/main.js"></script>
</body>
```


Non Semantic

VS
jeetsdev
Semantic
<header>
<section>
<article>
<article>
<aside>
<footer>

`<div>` tags?

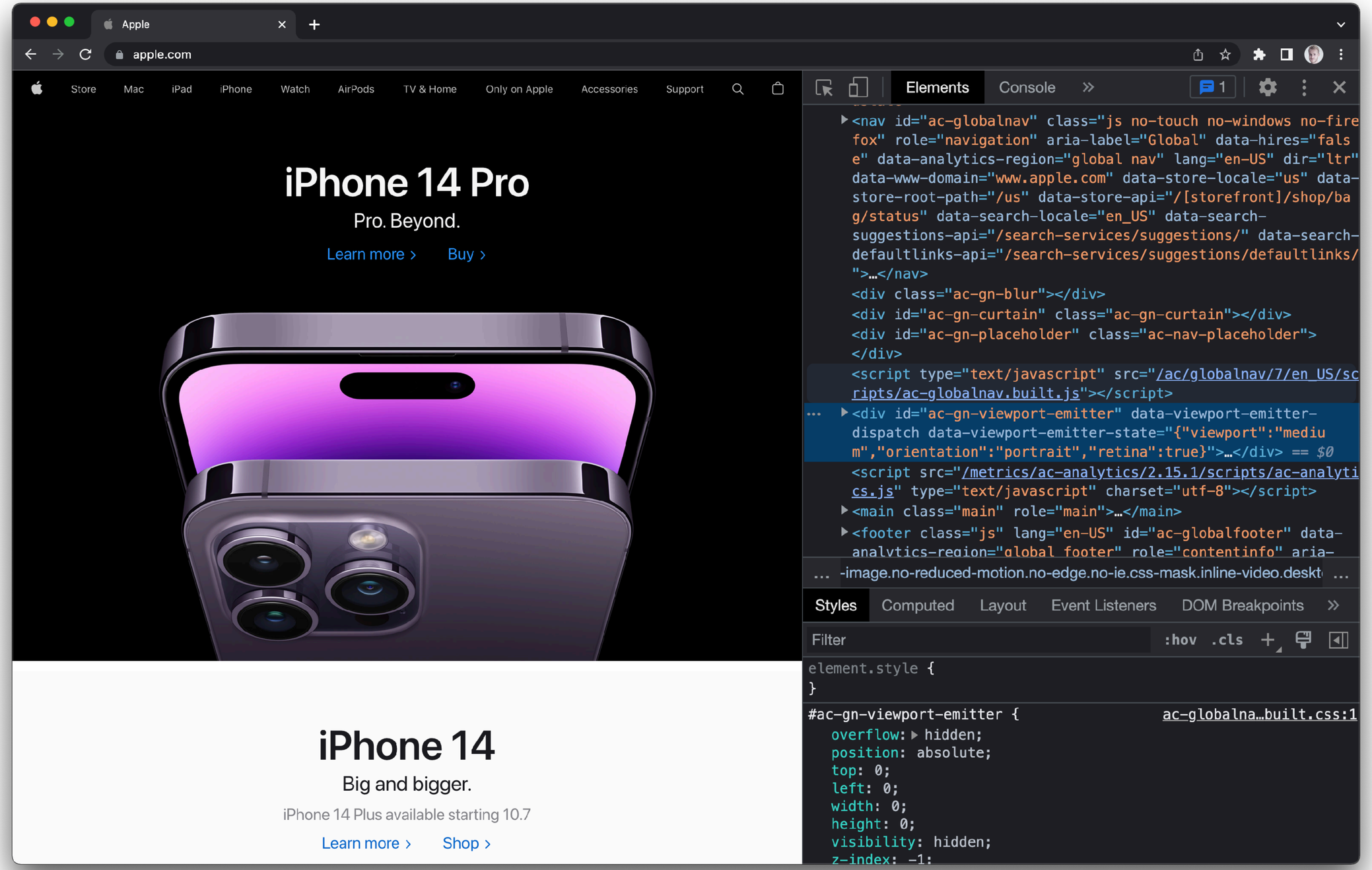
Many websites contain HTML code like:

```
<div id=“nav">
<div class=“header">
<div id="footer">
```

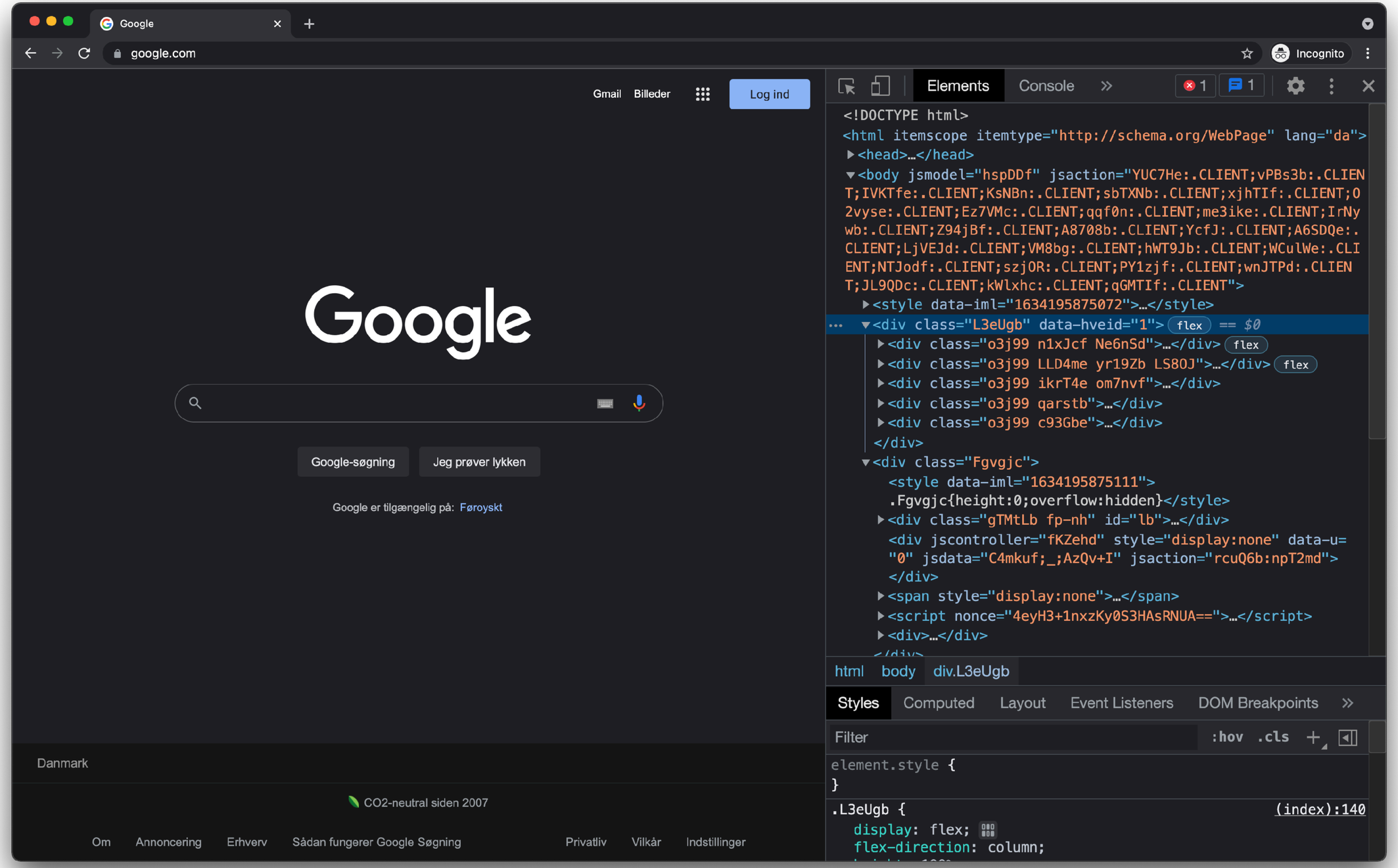
to indicate navigation, header, and footer.

Can we still use `div` tags?

https://www.w3schools.com/html/html5_semantic_elements.asp


Apple
apple.com
Store
Mac
iPad
iPhone
Watch
AirPods
TV & Home
Only on Apple
Accessories
Support
iPhone 14 Pro
Pro. Beyond.
Learn more ›
Buy ›
iPhone 14
Big and bigger.
iPhone 14 Plus available starting 10.7
Learn more ›
Shop ›
Elements
Console
<nav id="ac-globalnav" class="js no-touch no-windows no-firefox" role="navigation" aria-label="Global" data-hires="false" data-analytics-region="global nav" lang="en-US" dir="ltr" data-www-domain="www.apple.com" data-store-locale="us" data-store-root-path="/us" data-store-api="/[storefront]/shop/bag/status" data-search-locale="en_US" data-search-suggestions-api="/search-services/suggestions/" data-search-defaultlinks-api="/search-services/suggestions/defaultlinks/">…</nav>

<script type="text/javascript" src="/ac/globalnav/7/en_US/scripts/ac-globalnav.built.js"></script>
<div id="ac-gn-viewport-emitter" data-viewport-emitter-dispatch data-viewport-emitter-state="{"viewport":"medium","orientation":"portrait","retina":true}">… == $0
<script src="/metrics/ac-analytics/2.15.1/scripts/ac-analytics.js" type="text/javascript" charset="utf-8"></script>
<main class="main" role="main">…</main>
<footer class="js" lang="en-US" id="ac-globalfooter" data-analytics-region="global footer" role="contentinfo" aria-
-image.no-reduced-motion.no-edge.no-ie.css-mask.inline-video.deskt
Styles
Computed
Layout
Event Listeners
DOM Breakpoints
Filter
:hov .cls
element.style {
}
#ac-gn-viewport-emitter {
ac-globalna…built.css:1
overflow: hidden;
position: absolute;
top: 0;
left: 0;
width: 0;
height: 0;
visibility: hidden;
z-index: -1;


Google
google.com
Incognito
Gmail
Billeder
Log ind
Google
Google-søgning
Jeg prøver lykken
Google er tilgængelig på: Føroyskt
Danmark
CO2-neutral siden 2007
Om
Annoncering
Erhverv
Sådan fungerer Google Søgning
Privatliv
Vilkår
Indstillinger
Elements
Console
<!DOCTYPE html>
<html itemscope itemtype="http://schema.org/WebPage" lang="da">
<head>…</head>
<body jsmodel="hspDDf" jsaction="YUC7He:.CLIENT;vPBs3b:.CLIENT;IVKTfe:.CLIENT;KsNBn:.CLIENT;sbTXNb:.CLIENT;xjhTIf:.CLIENT;O2vyse:.CLIENT;Ez7VMc:.CLIENT;qqf0n:.CLIENT;me3ike:.CLIENT;IrNywb:.CLIENT;Z94jBf:.CLIENT;A8708b:.CLIENT;YcfJ:.CLIENT;A6SDQe:.CLIENT;LjVEJd:.CLIENT;VM8bg:.CLIENT;hWT9Jb:.CLIENT;WCulWe:.CLIENT;NTJodf:.CLIENT;szjOR:.CLIENT;PY1zjf:.CLIENT;wnJTPd:.CLIENT;JL9QDc:.CLIENT;kWlxhc:.CLIENT;qGMTIf:.CLIENT">
<style data-iml="1634195875072">…</style>

…
…
…
…
…

<style data-iml="1634195875111">
.Fgvgjc{height:0;overflow:hidden}</style>
…

…
<script nonce="4eyH3+1nxzKy0S3HAsRNUA==">…</script>
…
html
body
div.L3eUgb
Styles
Computed
Layout
Event Listeners
DOM Breakpoints
Filter
:hov
.cls
element.style {
}
.L3eUgb {
(index):140
display: flex;
flex-direction: column;

HTML Semantic Elements

w3schools.com/html/html5_semantic_elements.asp

Incognito

Tutorials References Exercises Website Paid Courses Log in

HTML CSS JAVASCRIPT SQL PYTHON PHP BOOTSTRAP

HTML Semantic Elements

❮ Previous Next ❯

Semantic elements = elements with a meaning.

What are Semantic Elements?

A semantic element clearly describes its meaning to both the browser and the developer.

Examples of **non-semantic** elements: `<div>` and `<span>` - Tells nothing about its content.

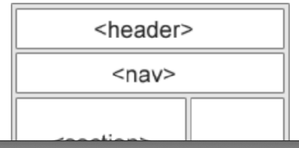
Examples of **semantic** elements: `<form>`, `<table>`, and `<article>` - Clearly defines its content.

Semantic Elements in HTML

Many web sites contain HTML code like: to indicate navigation, header, and footer.

In HTML there are some semantic elements that can be used to define different parts of a web page:

- <article>
- <aside>
- <details>
- <figcaption>



Elements Console 2 2 28

```
<body> == $0
  <div class="w3-bar w3-card-2 notranslate" style="position: relative;z-index: 4;font-size: 18px;background-color: white;color:#282A35;padding-left:12px;padding-right:16px;font-family: 'Source Sans Pro', sans-serif;">…</div>
  <div style="display:none;position:absolute;z-index:4;right:52px;height:44px;background-color:#282A35;letter-spacing:normal;" id="googleSearch">…</div>
  <div style="display:none;position:absolute;z-index:3;right:111px;height:44px;background-color:#282A35;text-align:right;padding-top:9px;" id="google_translate_element"></div>
  <div class="w3-card-2 topnav notranslate" id="topnav" style="position: relative;">…</div>
  <div id="myAccordion" class="w3-card-2 w3-center w3-hide-large w3-hide-medium" style="width: 100%; position: absolute; display: none; background-color: rgb(231, 233, 235); padding-top: 0px;">…</div>
  <script> </script>
  <div class="w3-sidebar w3-collapse" id="sidenav" style="top: 118px; display: none;">…</div>
  <div class="w3-main w3-light-grey" id="belowtopnav" style="margin-left: 220px; padding-top: 0px;">
    <div class="w3-row w3-white">
      ::before
      <div class="w3-col l10 m12" id="main">
        <div id="mainLeaderboard" style="overflow:hidden;">…</div>
        <h1>…</h1>
        <div class="w3-clear nextprev"> </div>
```

html body

Styles Computed Layout Event Listeners DOM Breakpoints

Filter :hov .cls

```
element.style {
}
html, body {                                    w3schools28.css:27
  font-family: Verdana,sans-serif;
  font-size: 15px;
```

tags?

Use semantic elements as much as possible (!)

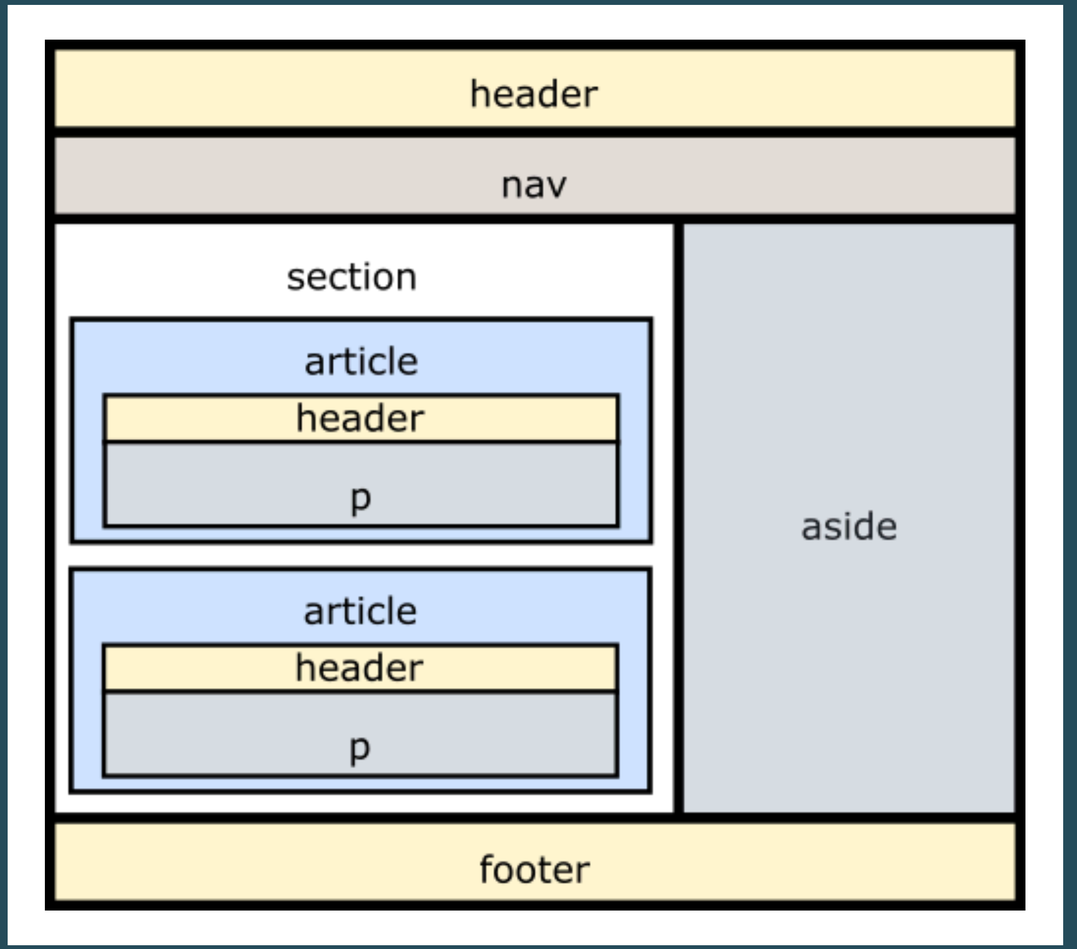
... but don't let it stop you from building something amazing 🎉

- RACE

https://www.w3schools.com/html/html5_semantic_elements.asp

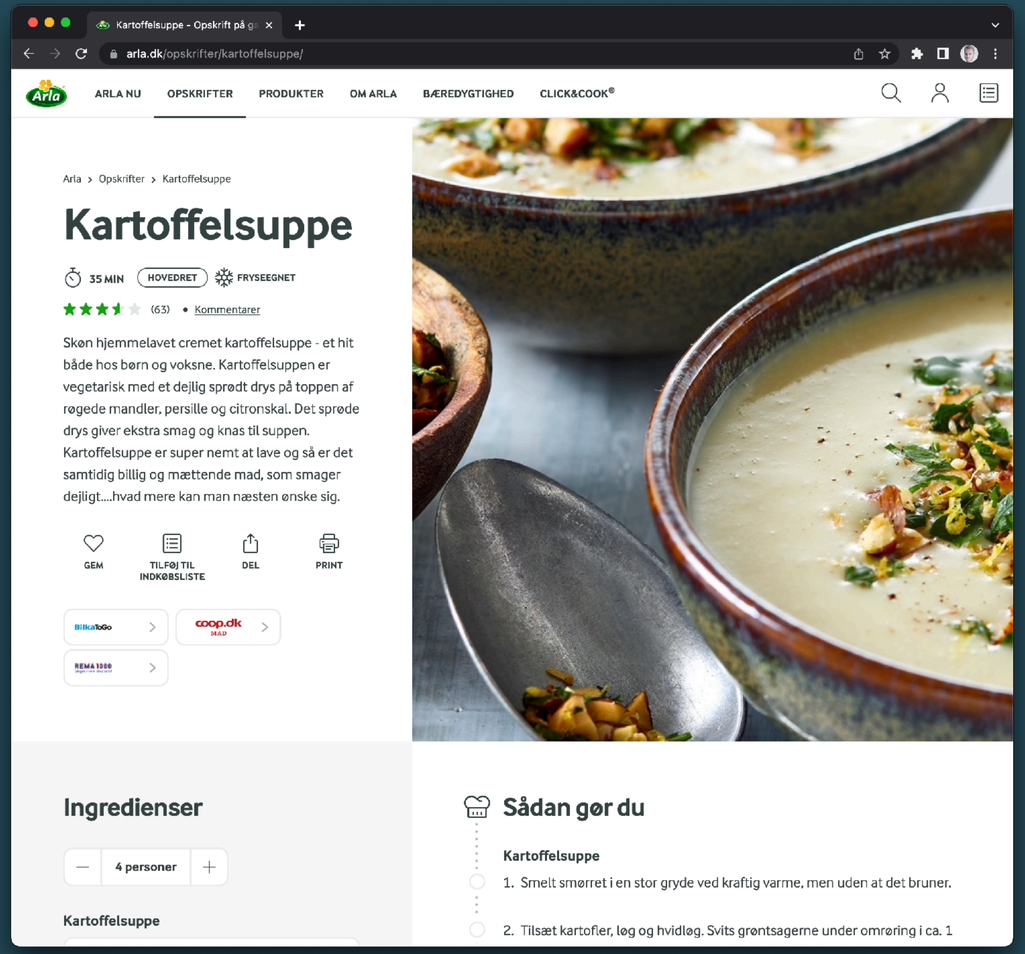
HTML Semantics Exercise

1. Use the `project-template` from GitHub. Make a copy of the project and paste the project folder into your own local dev folder.
2. Create a structure matching the image to the right. Use HTML Semantic tags.
3. Add content to the structure, like headings, links, text (paragraphs), images and contact info. Explore and use HTML Element Reference & HTML Semantics.
4. Extra: Add some styling (CSS).



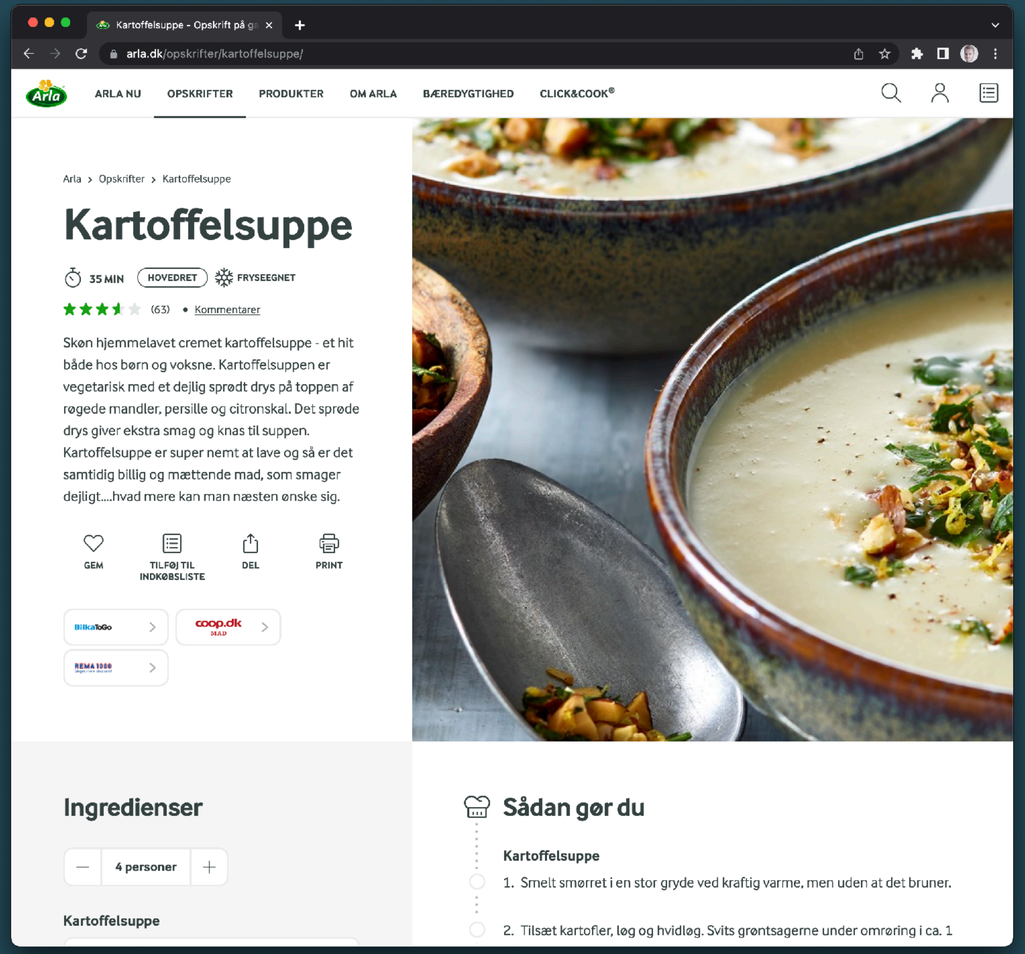
HTML Recipe Exercise

- Tag udgangspunkt i dit projekt fra foregående opgave (tag eventuelt en kopi).
- Strukturér en opskriftsside med elementer som overskrift(er), beskrivelse(r), billede(r), liste af ingredienser og fremgangsmåde. Du skal bruge HTML elementer og semantiske tags.
- Fokus er struktur i HTML'en - ikke styling og layout.
- Du kan bruge elementer som h1, h2, h3, p, section, article, aside, main, ul, li og mange andre. Gå på opdagelse i "dokumentationen".
- Brug indhold fra websitet til venstre eller en anden opskrift efter eget ønske.

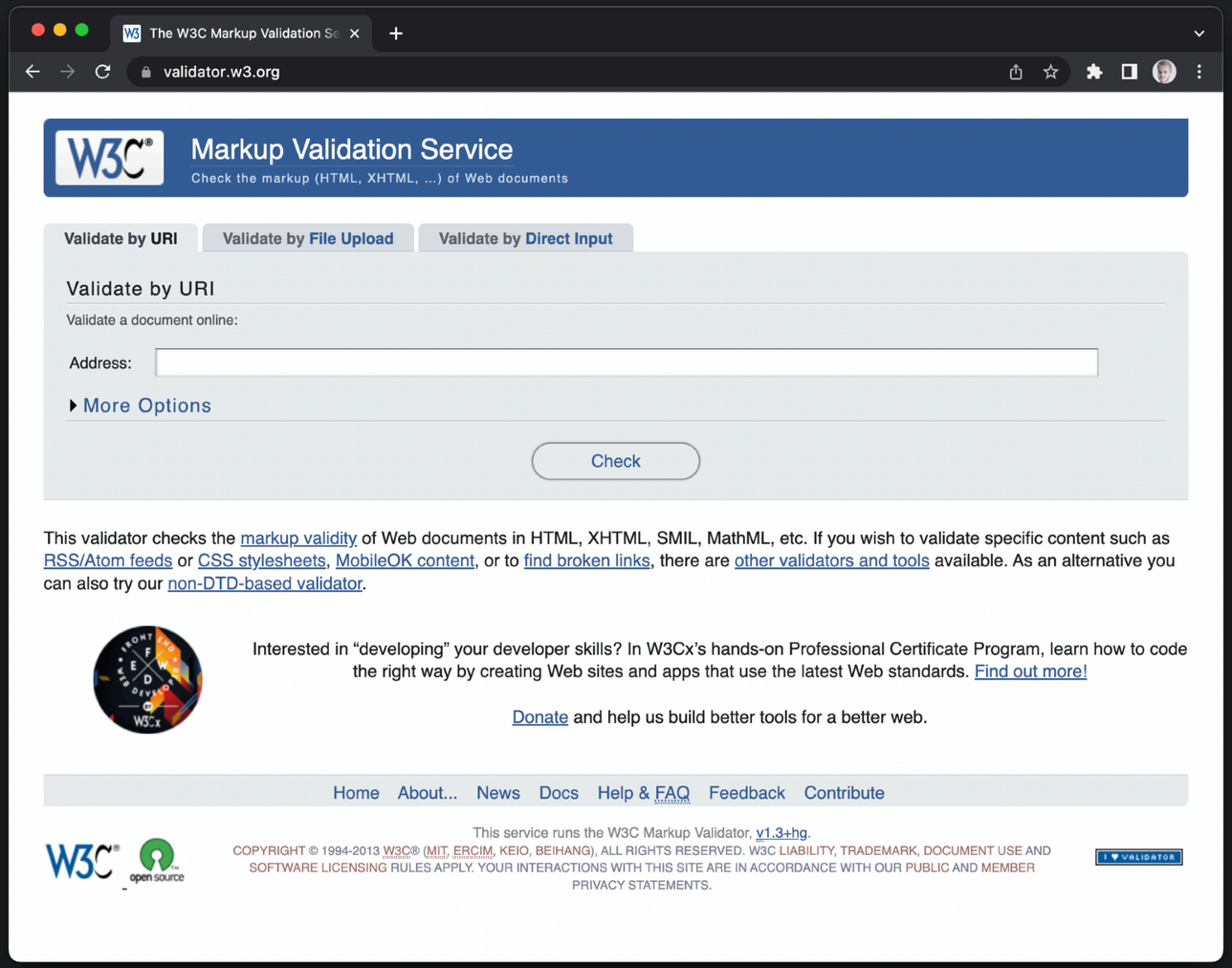


HTML Recipe Exercise

- Undersøg Arla-opskrifter.
- Er HTML'en semantisk?
  Hvorfor, hvorfor ikke?
- Forbederingsforslag?

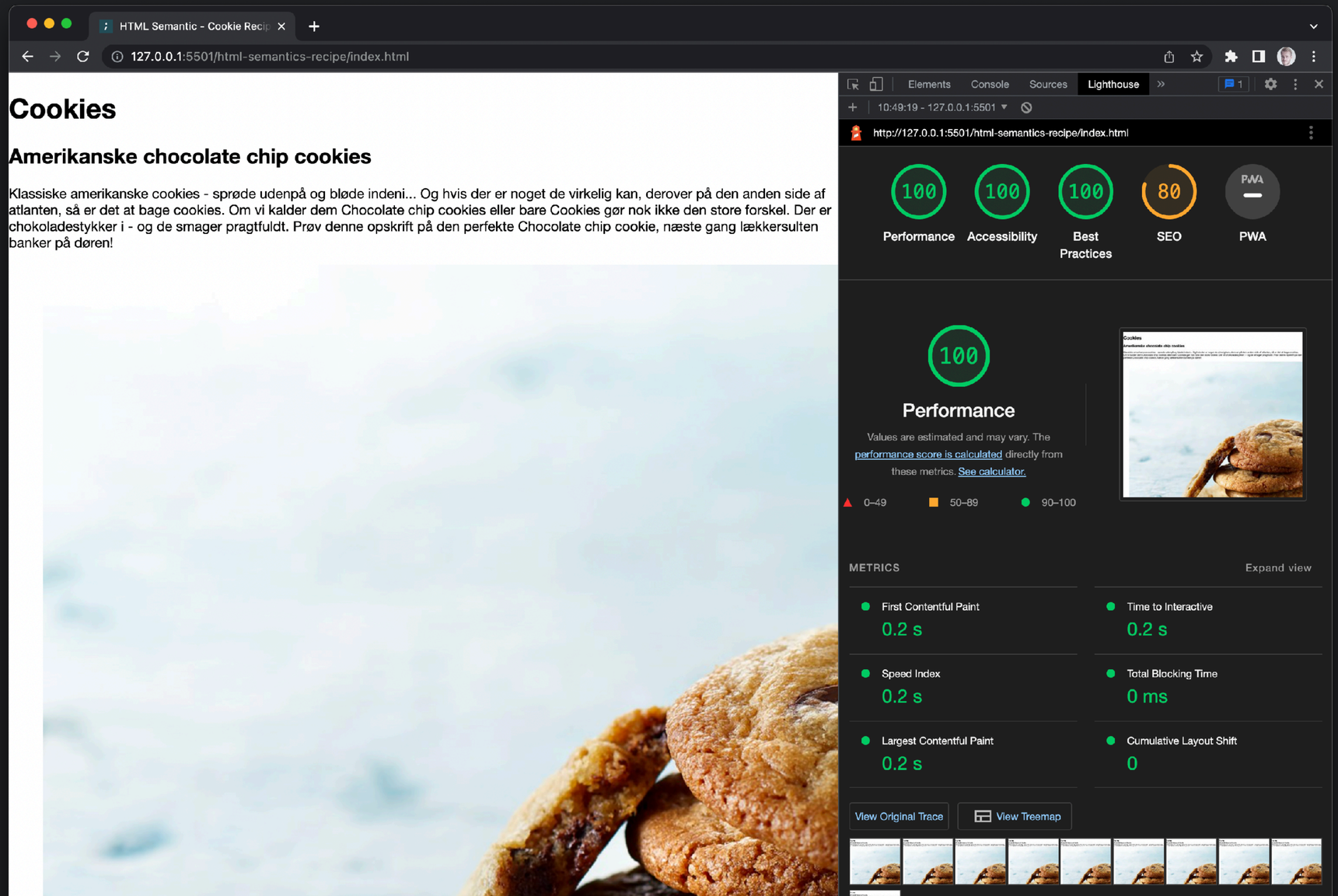


Validate your HTML



https://validator.w3.org/

Lighthouse in Chrome



Semantic HTML & SEO (indexability)

Why tell the browser what the HTML elements represents?

... to improve

Search Engine Optimization

Tell what your content is about and

by hierarchy, tell what’s important

Why Semantic HTML?

“Because semantic HTML uses elements for their given purpose, it’s easier for both people and machines to read and understand it.”

5 Ways to Write Semantic HTML and Improve Webpage SEO and Accessibility

“Semantic HTML means using correct HTML elements for their correct purpose as much as possible. Semantic elements are elements with a meaning; if you need a button, use the `<button>` element (and not a `<div>` element).”

Semantic

```
<button>Report an Error</button>
```

Non-semantic

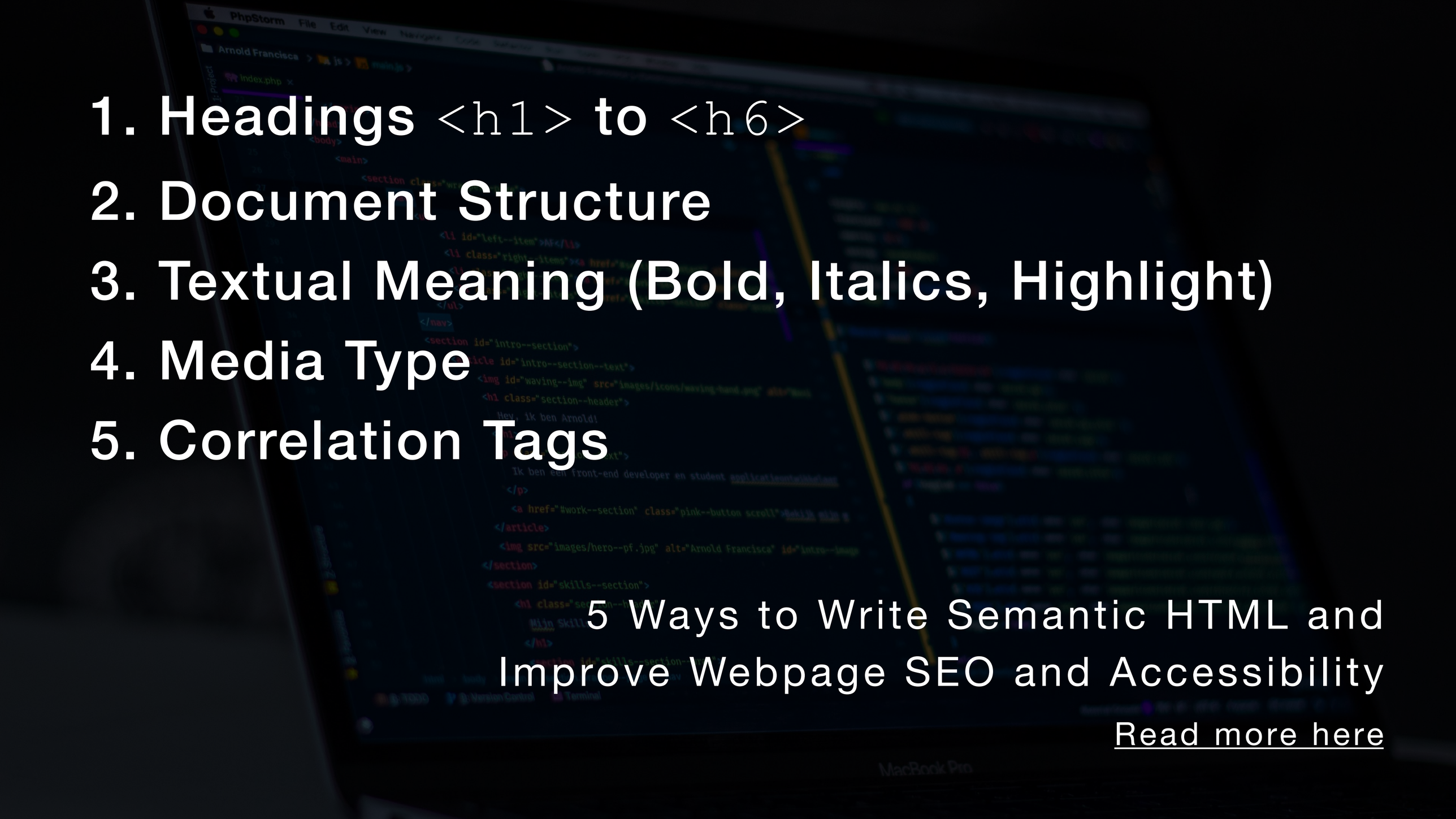
```
<div>Report an Error</div>
```

https://www.w3schools.com/html/html_accessibility.asp

Examples of non-semantic elements: `<div>` and `<span>`
- Tells nothing about its content.

Examples of semantic elements: `<form>`, `<table>`, and `<article>`
- Clearly defines its content.

https://www.w3schools.com/html/html_accessibility.asp



1. Headings `<h1>` to `<h6>`
2. Document Structure
3. Textual Meaning (Bold, Italics, Highlight)
4. Media Type
5. Correlation Tags

5 Ways to Write Semantic HTML and Improve Webpage SEO and Accessibility

Read more here

What’s Accessibility?

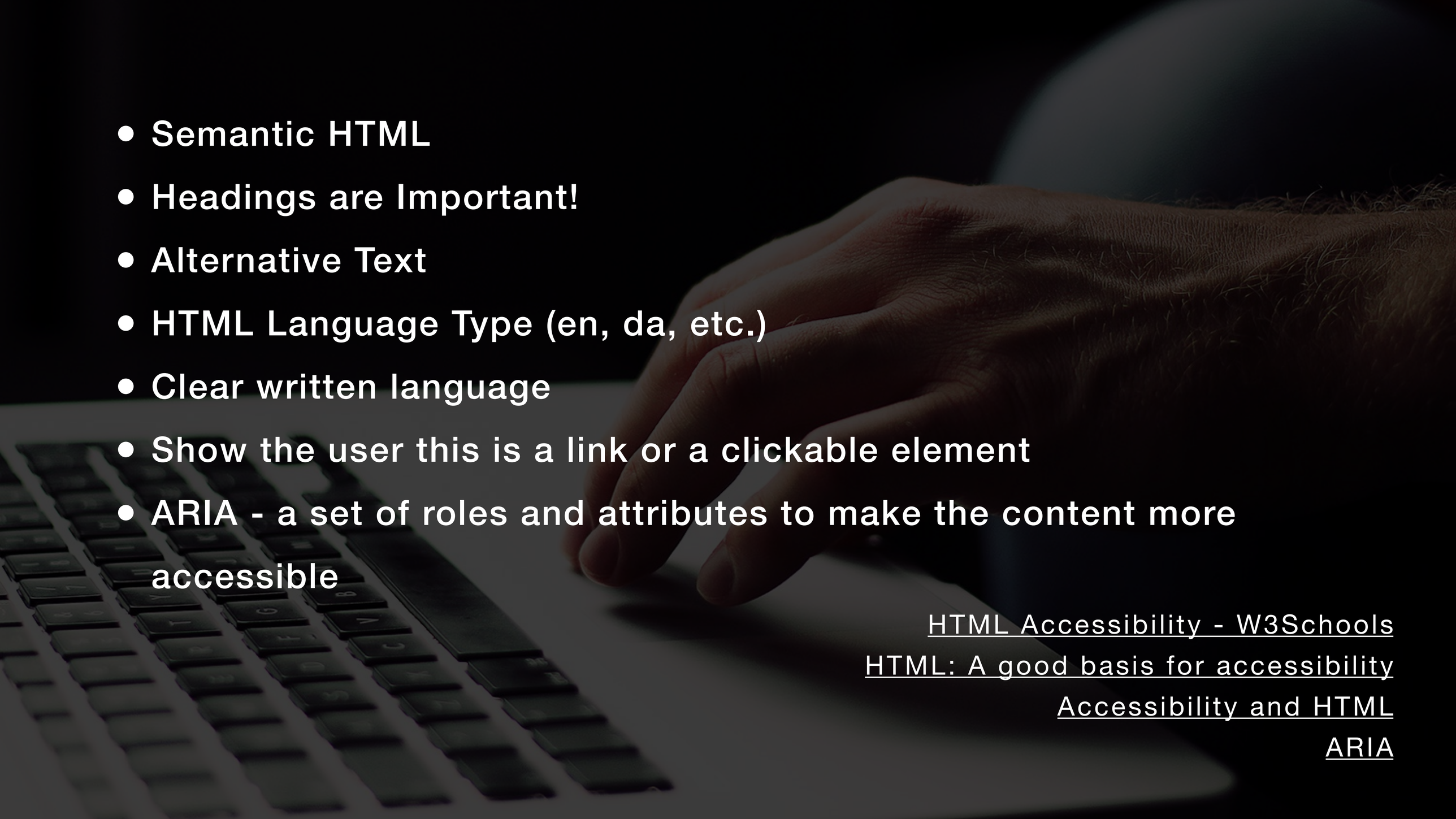
“[...] the practice of making your websites usable by as many people as possible.”

- and machines, programs and devices.

What is accessibility?

“Web accessibility means that people with disabilities can use the Web.”

Why Web Accessibility Is Important and How You Can Accomplish It



- Semantic HTML
- Headings are Important!
- Alternative Text
- HTML Language Type (en, da, etc.)
- Clear written language
- Show the user this is a link or a clickable element
- ARIA - a set of roles and attributes to make the content more accessible

HTML Accessibility - W3Schools

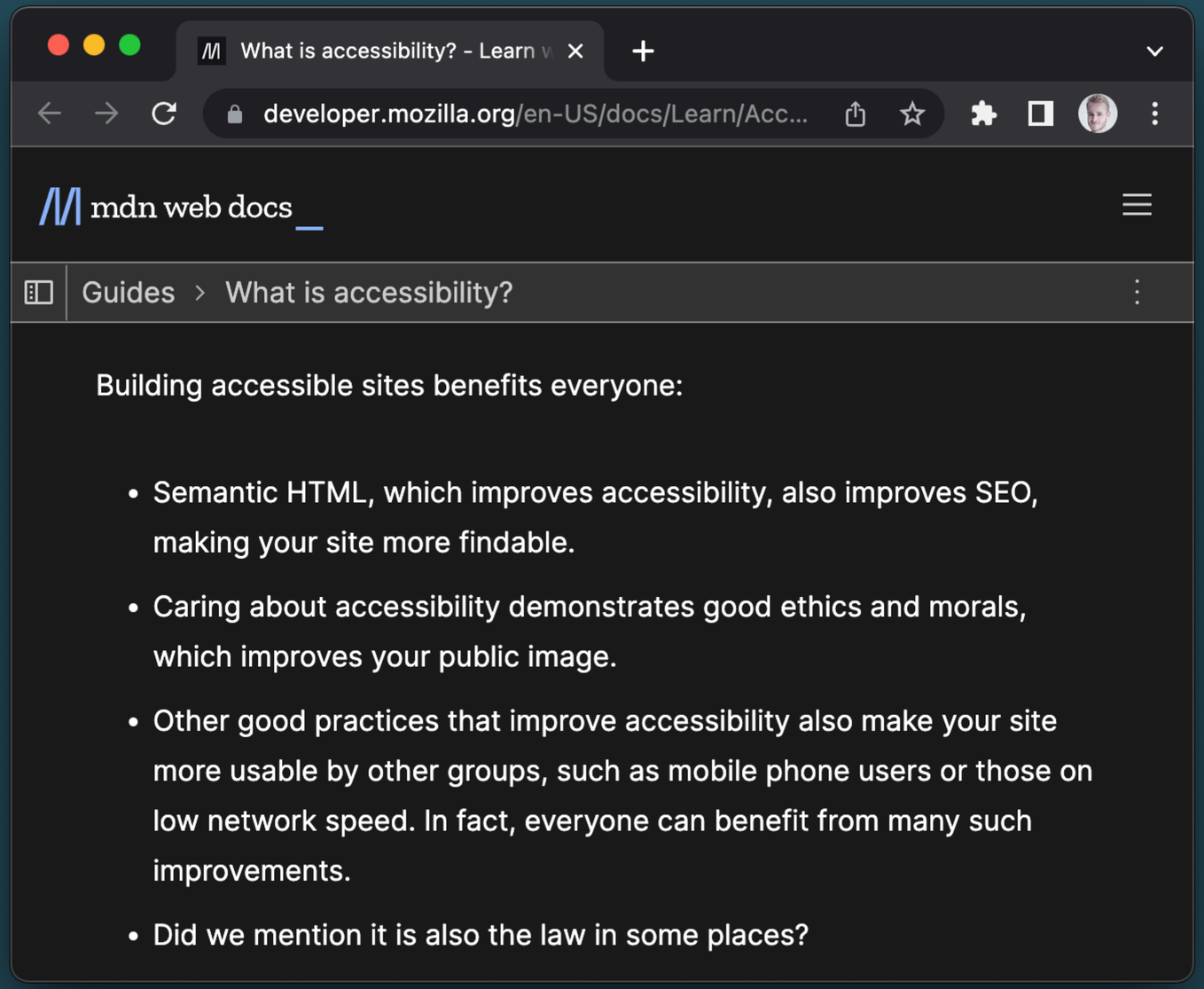
HTML: A good basis for accessibility

Accessibility and HTML

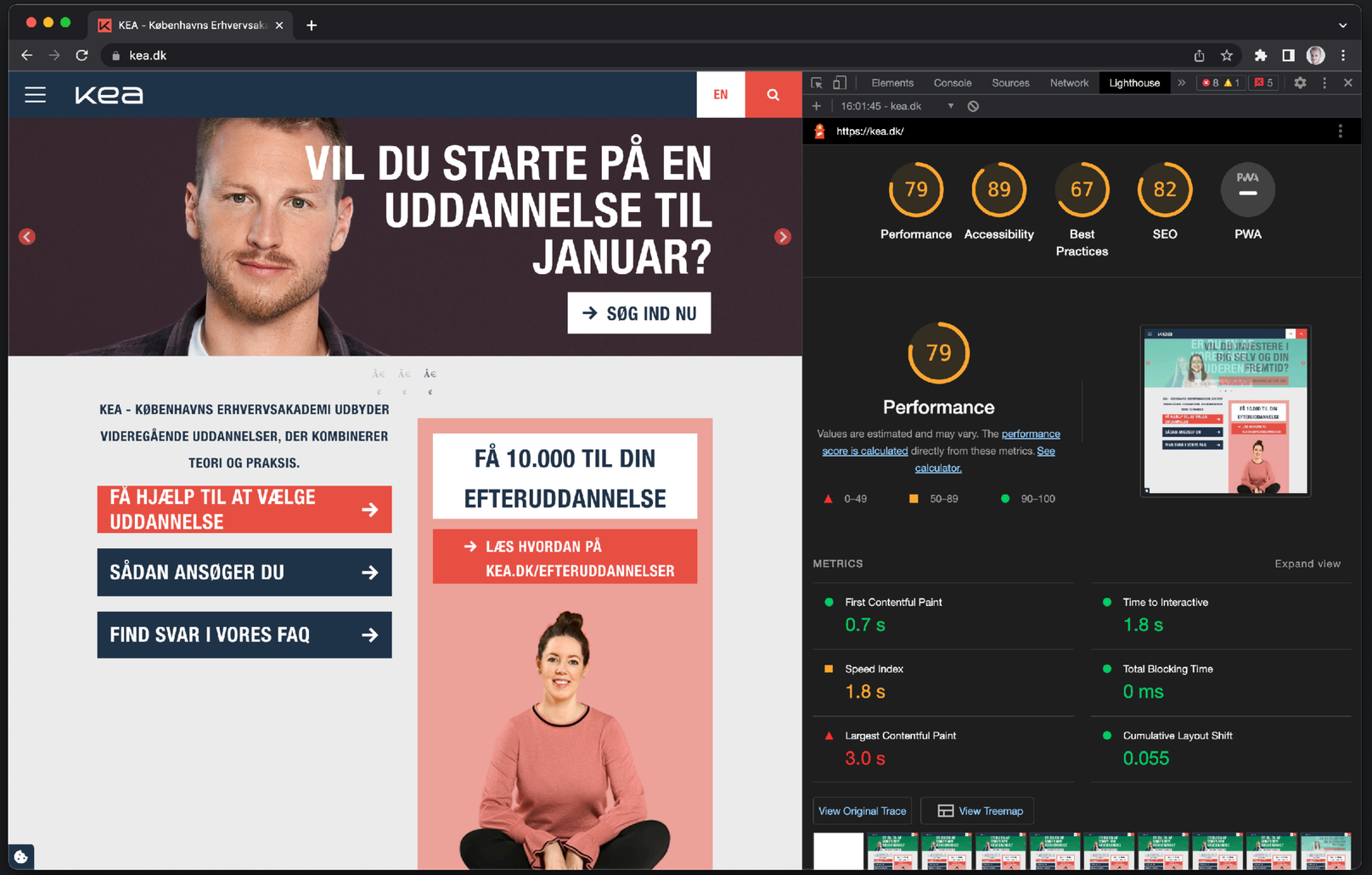
ARIA

The benefits of accessibility?

What is accessibility?



Lighthouse in Chrome



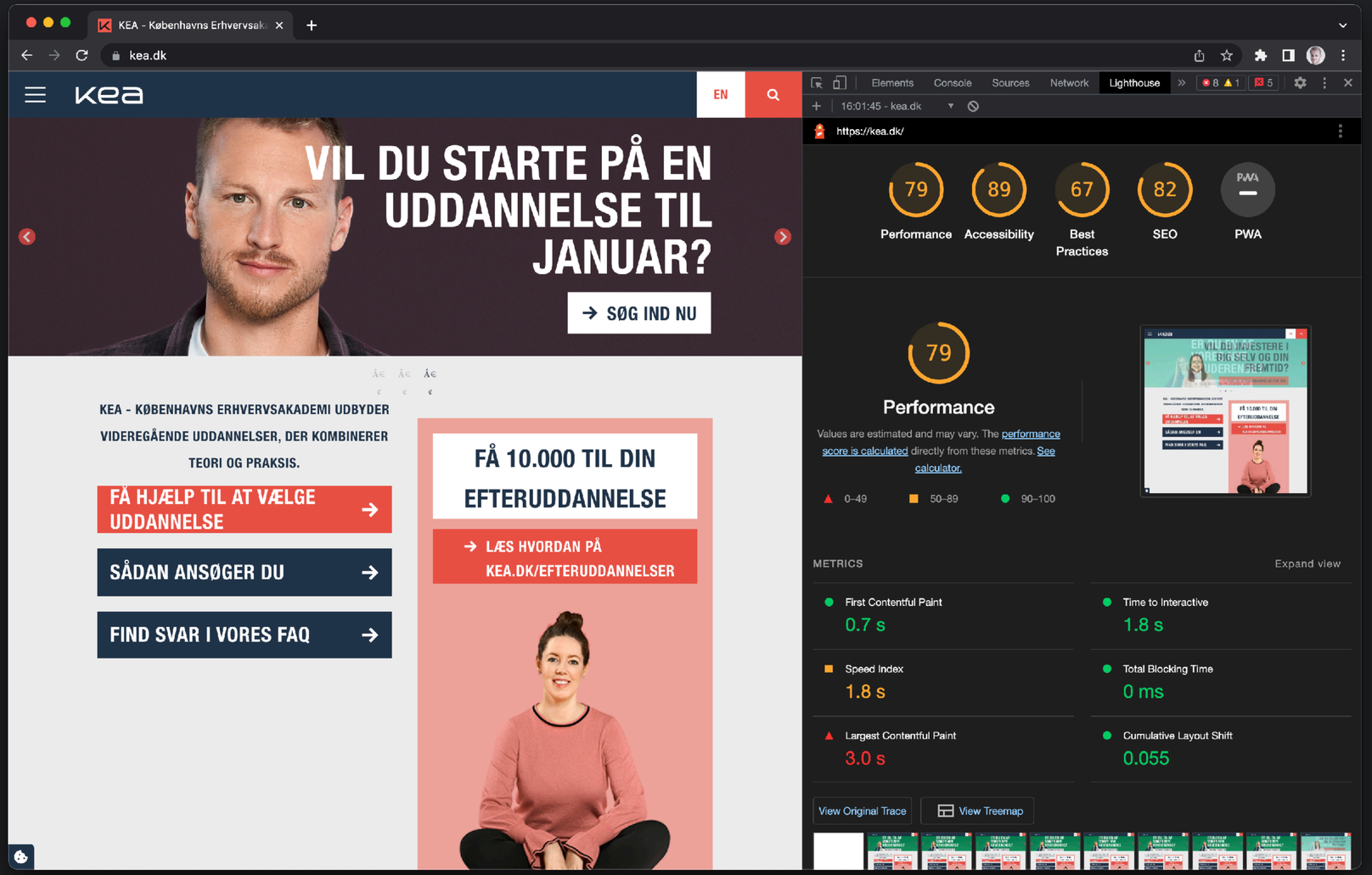
Performance is about...

- Load time of your website
- Is your site usable while loading resources? (Lazy loading)
- Smoothness and interactivity
- Does your site seem fast to the user? (Perceived performance)
- Measuring performance

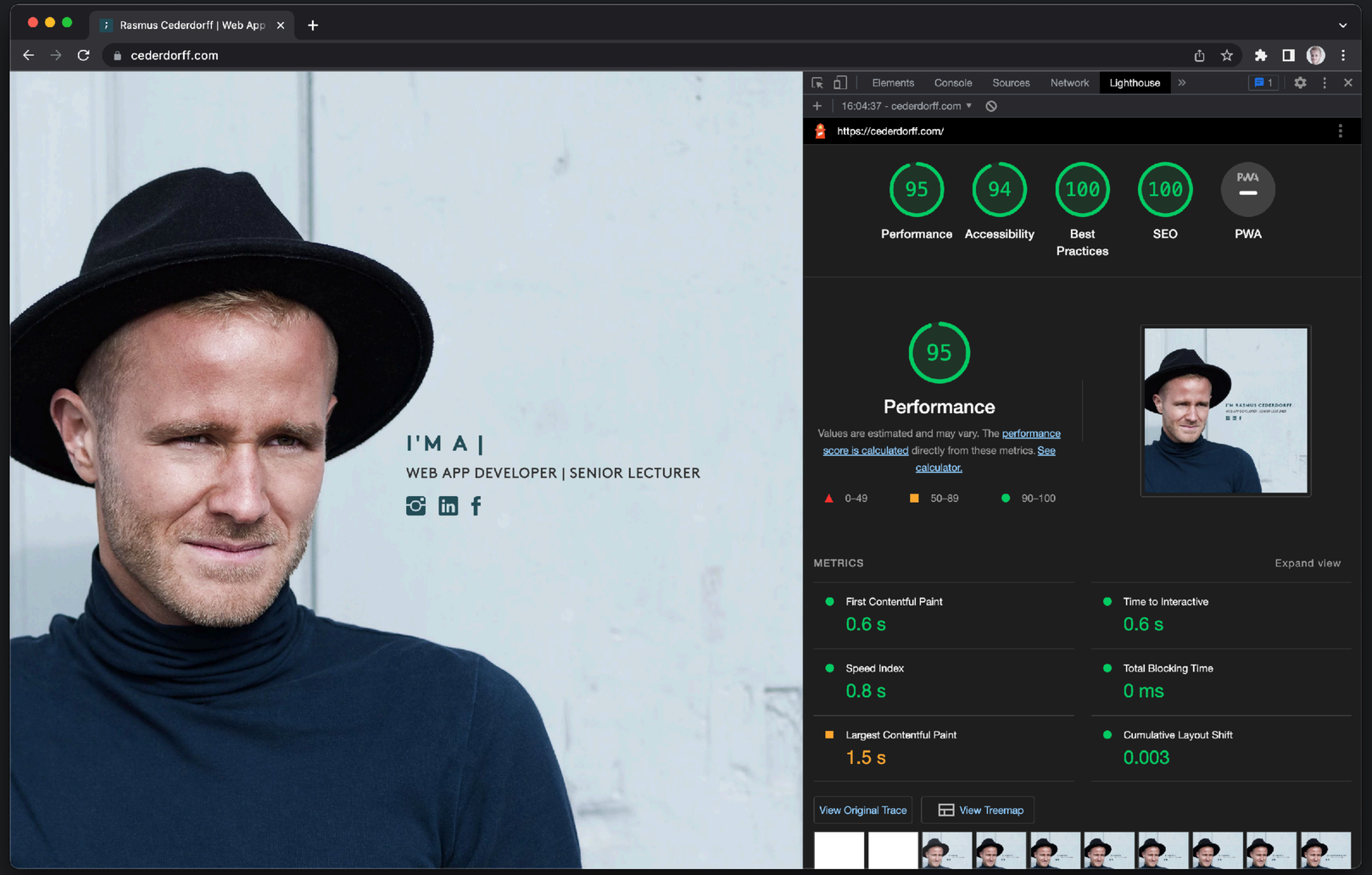
Reduce and minimise size (ex images), loading and response time

What is web performance?

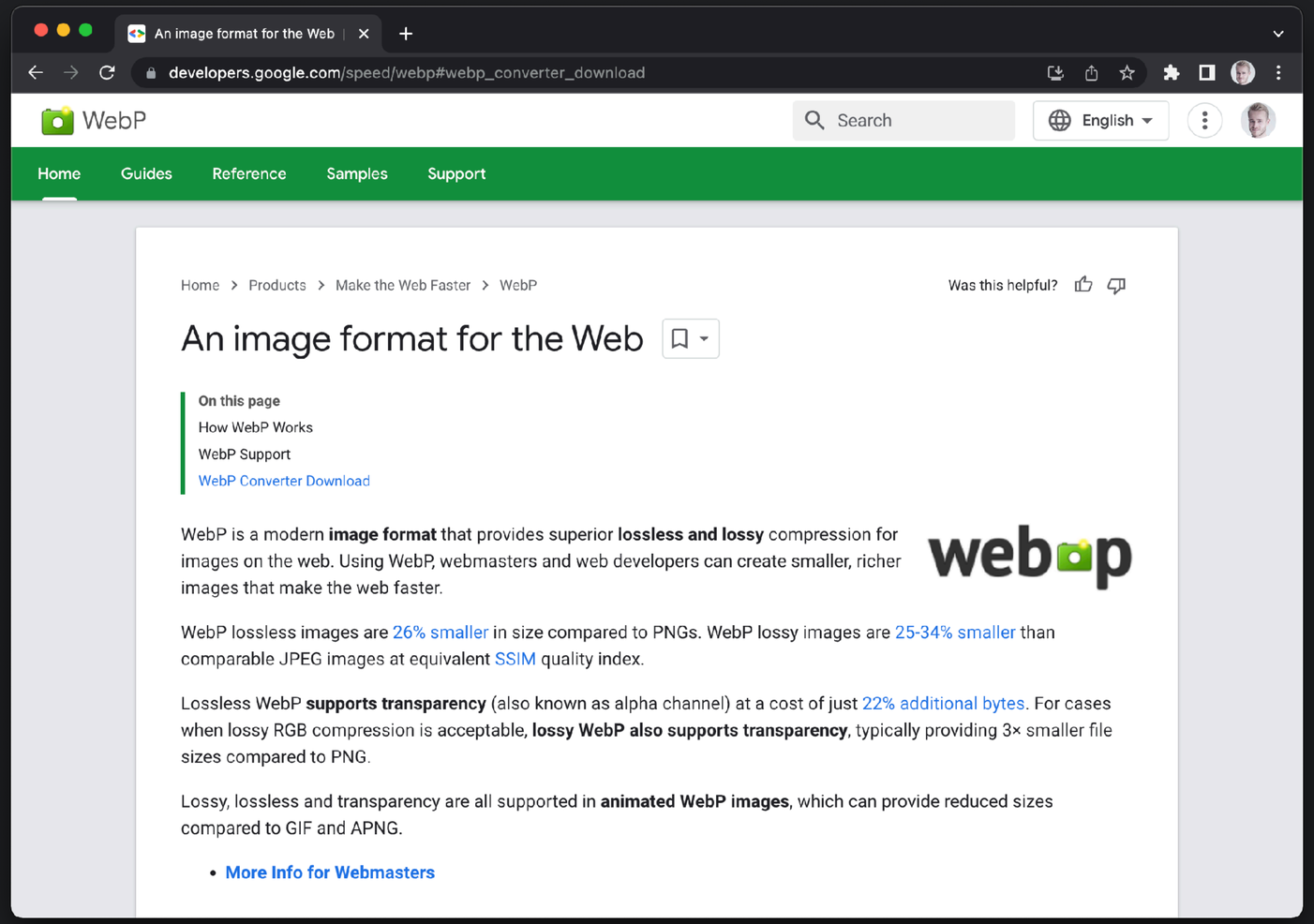
Lighthouse in Chrome



Not bragging 🤷

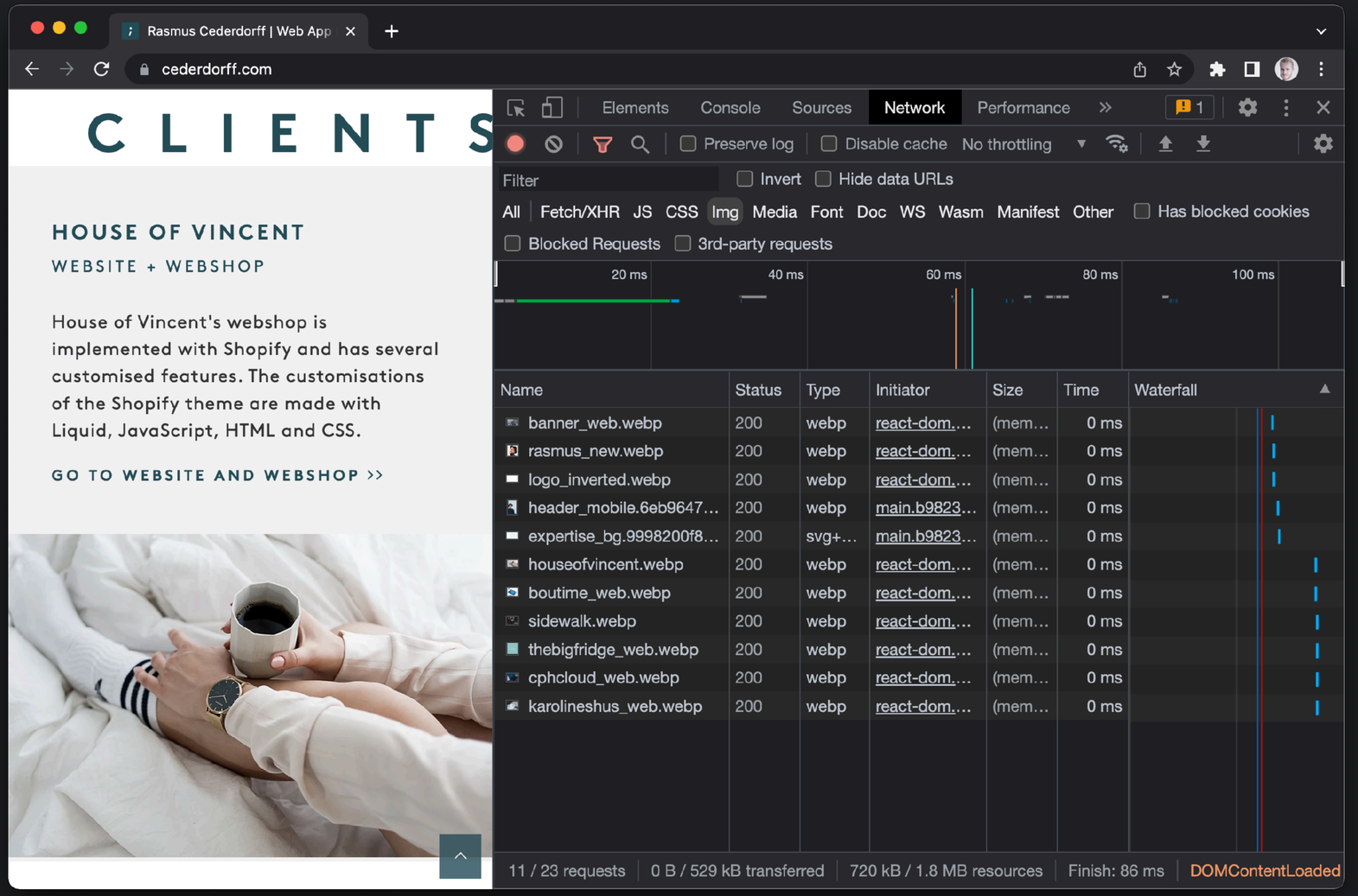


Images, sizes & formats



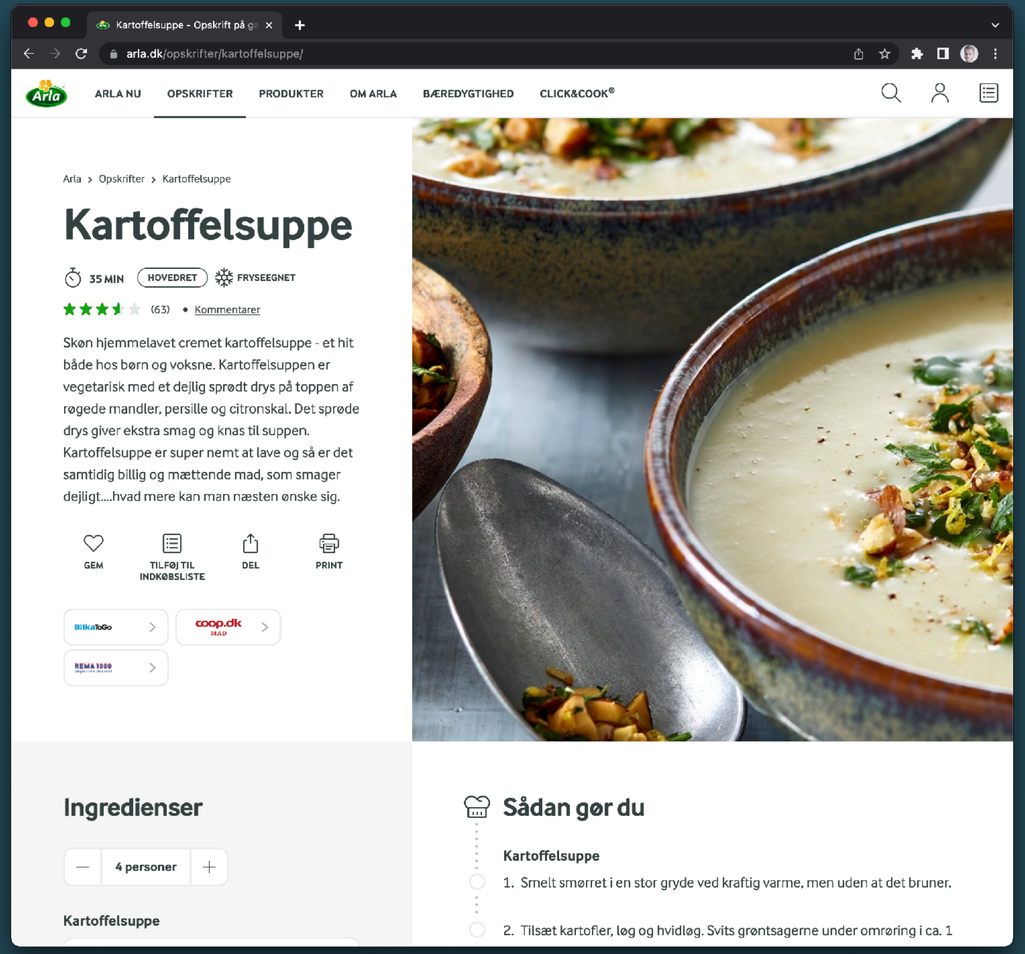
https://developers.google.com/speed/webp

.webp


Rasmus Cederdorff | Web App
cederdorff.com
CLIENTS
HOUSE OF VINCENT
WEBSITE + WEBSHOP
House of Vincent's webshop is implemented with Shopify and has several customised features. The customisations of the Shopify theme are made with Liquid, JavaScript, HTML and CSS.
GO TO WEBSITE AND WEBSHOP >>
Elements Console Sources Network Performance
Preserve log Disable cache No throttling
Filter Invert Hide data URLs
All Fetch/XHR JS CSS Img Media Font Doc WS Wasm Manifest Other Has blocked cookies
Blocked Requests 3rd-party requests
20 ms 40 ms 60 ms 80 ms 100 ms
Name Status Type Initiator Size Time Waterfall
banner_web.webp 200 webp react-dom.... (mem... 0 ms
rasmus_new.webp 200 webp react-dom.... (mem... 0 ms
logo_inverted.webp 200 webp react-dom.... (mem... 0 ms
header_mobile.6eb9647... 200 webp main.b9823... (mem... 0 ms
expertise_bg.9998200f8... 200 svg+... main.b9823... (mem... 0 ms
houseofvincent.webp 200 webp react-dom.... (mem... 0 ms
boutime_web.webp 200 webp react-dom.... (mem... 0 ms
sidewalk.webp 200 webp react-dom.... (mem... 0 ms
thebigfridge_web.webp 200 webp react-dom.... (mem... 0 ms
cphcloud_web.webp 200 webp react-dom.... (mem... 0 ms
karolineshus_web.webp 200 webp react-dom.... (mem... 0 ms
11 / 23 requests 0 B / 529 kB transferred 720 kB / 1.8 MB resources Finish: 86 ms DOMContentLoaded

Webp Exercise

1. Find og download et par billeder du kan bruge på din opskriftsside.
2. Sørg for at billeder ikke er større end nødvendigt (opløsning, height og width).
3. Test med Lighthouse. Tag et screenshot af resultatet.
4. Konverter dine billeder til .wepb (brug fx en online converter).
5. Brug webp-filerne i stedet for dine tidligere billedfiler.
6. Test med Lighthouse igen og sammenlign resultatet med tidligere



WHAT IS CSS?

CSS stands for **C**ascading **S**tyle **S**heets

CSS describes how HTML elements are to be displayed on screen, paper, or in other media

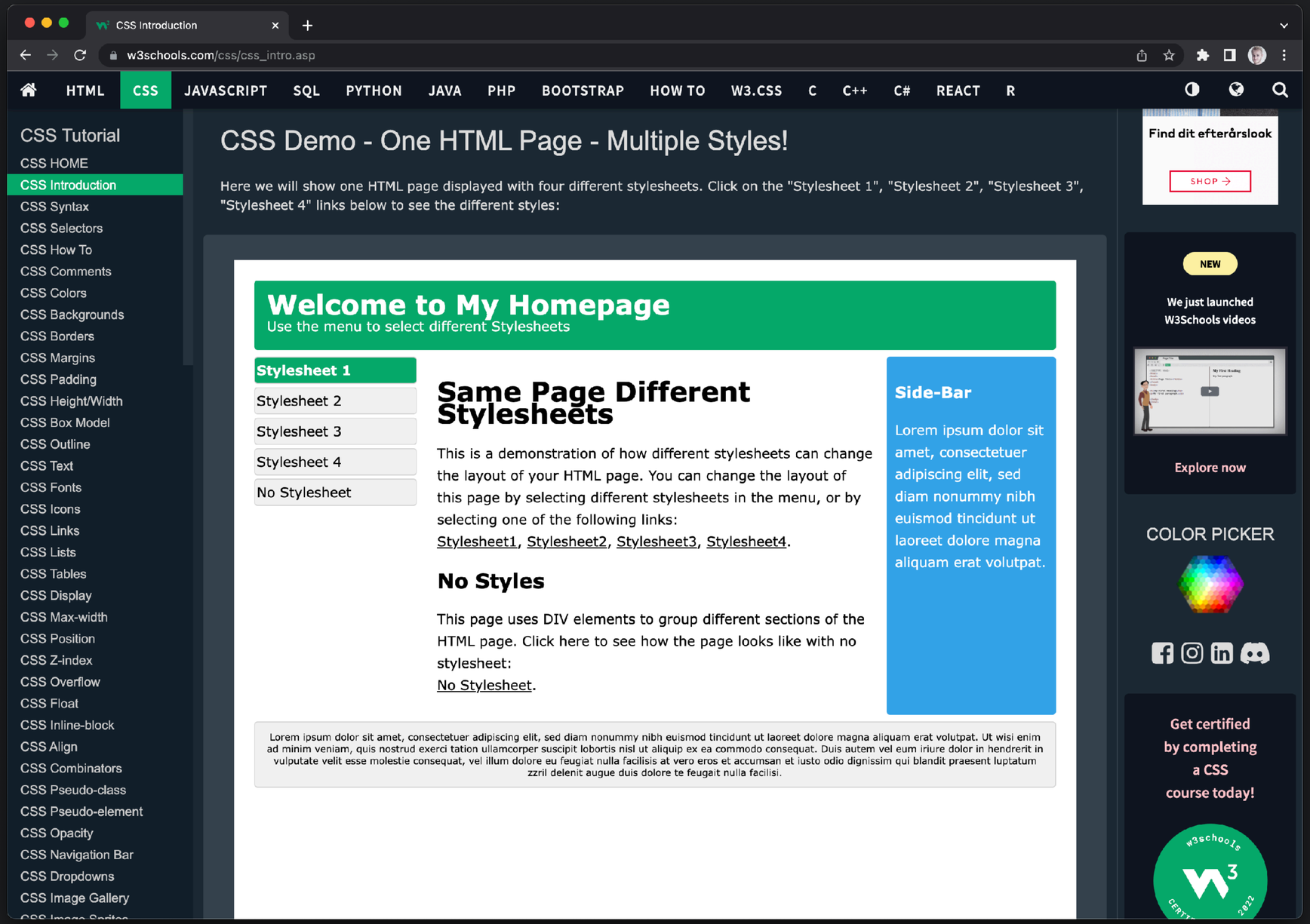
CSS **saves a lot of work**. It can control the layout of multiple web pages all at once

External stylesheets are stored in **CSS files**

https://www.w3schools.com/css/css_intro.asp

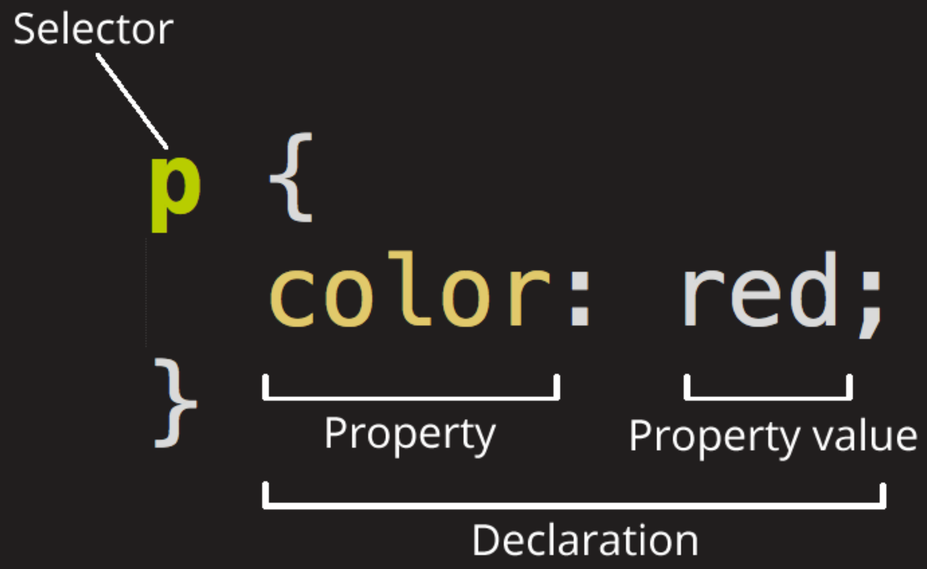
styles.css

```
1   body {
2     font-family: Helvetica, Arial, sans-serif;
3     text-align: center;
4   }
5
6   h1 {
7     font-weight: lighter;
8   }
9
10  #my-paragraph {
11    color: blue;
12  }
13
14  .big {
15    font-size: 25px;
16  }
17
```



https://www.w3schools.com/css/css_intro.asp

Anatomy of a CSS ruleset



- **Selector**: the HTML element(s) to be styled
- **Declaration**: single rule like `color: red` or `margin: 10px`
- **Properties**: ways you can style HTML element(s). Ex `color` or `margin`. You choose what properties you want to affect.
- **Property value**: One out of many possible appearances for a given style (property). Ex `red` or `10px`.

Read more: Anatomy of a CSS ruleset

index.html

```html
<!DOCTYPE html>
<html>
  <head>
    <title>Page Title</title>
    <link rel="stylesheet" href="styles.css" />
  </head>
  <body>
    <h1>This is a Heading</h1>
    <p>This is a paragraph.</p>
    <p id="my-paragraph">This is another paragraph.</p>
    <p class="big">This is another paragraph.</p>
  </body>
</html>
```

styles.css

```css
body {
  font-family: Helvetica, Arial, sans-serif;
  text-align: center;
}

h1 {
  font-weight: lighter;
}

#my-paragraph {
  color: blue;
}

.big {
  font-size: 25px;
}
```

Browser Tests

https://9nypo.csb.app/

This is a Heading

This is a paragraph.

This is another paragraph.

This is another paragraph.

index.html

```html
<!DOCTYPE html>
<html>
  <head>
    <title>Page Title</title>
    <link rel="stylesheet" href="styles.css" />
  </head>
  <body>
    <h1>This is a Heading</h1>
    <p>This is a paragraph.</p>
    <p id="my-paragraph">This is another paragraph.</p>
    <p class="big">This is another paragraph.</p>
    <img src="https://cederdorff.com/img/karolineshus_web.jpg" />
  </body>
</html>
```

styles.css

```css

#my-paragraph {
  color: blue;
}

.big {
  font-size: 25px;
}

img {
  width: 100%;
}

```

Browser Tests

https://9nypo.csb.app/

This is a Heading

This is a paragraph.

This is another paragraph.

This is another paragraph.

CSS SELECTORS

CSS selectors are used to "find" (or select) the HTML elements you want to style.

Simple selectors (name, id, class)

Combinator selectors (a specific relationship between them)

Pseudo-class selectors (a certain state)

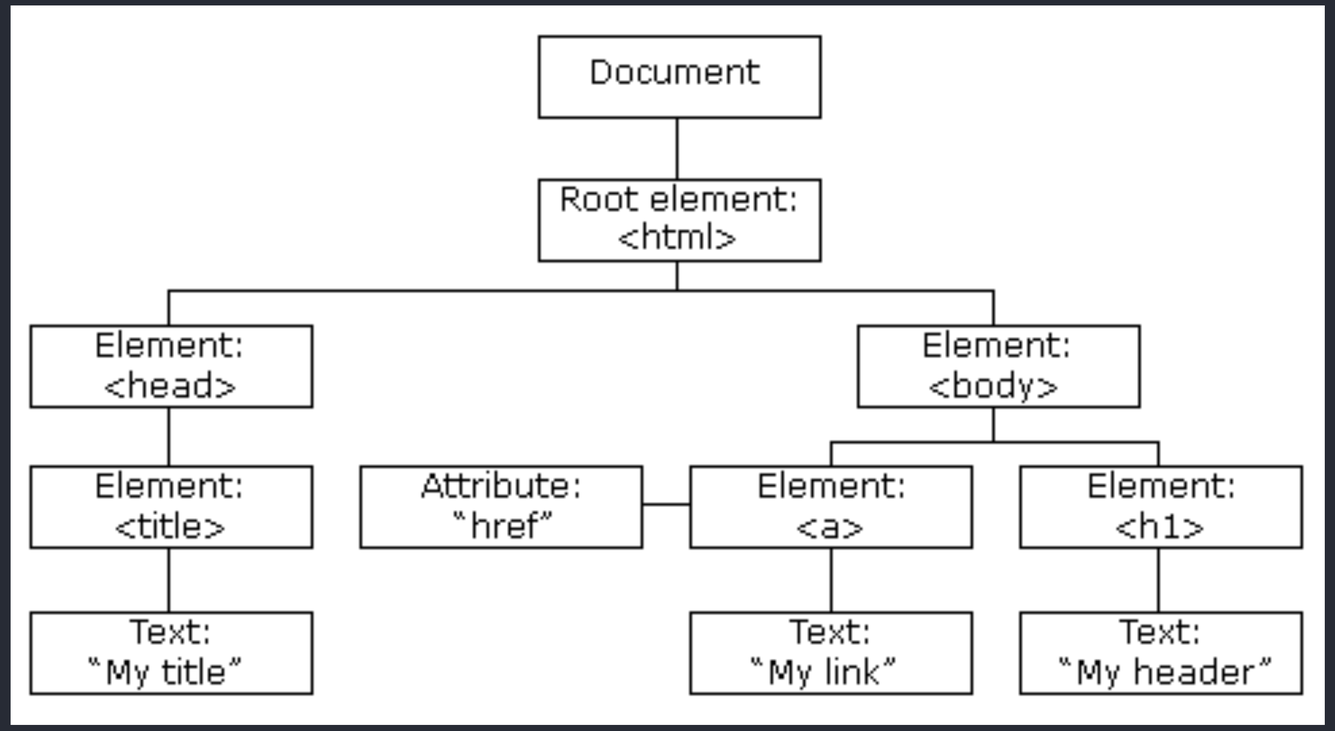
Pseudo-elements selectors (a part of an element)

Attribute selectors (attribute or attribute value)

https://www.w3schools.com/css/css_selectors.asp

The HTML DOM (Document Object Model)

A model as a tree of Objects

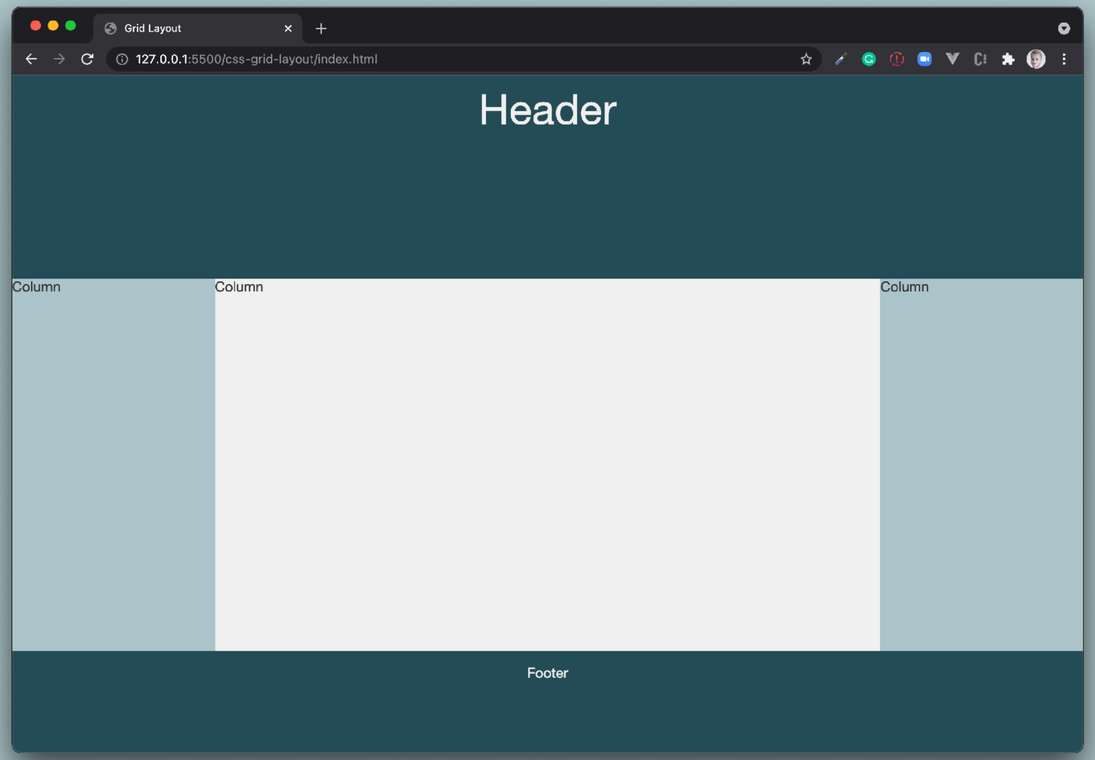


- Finding/ select HTML Elements:
  - id
  - tag name
  - class name
  - CSS selectors

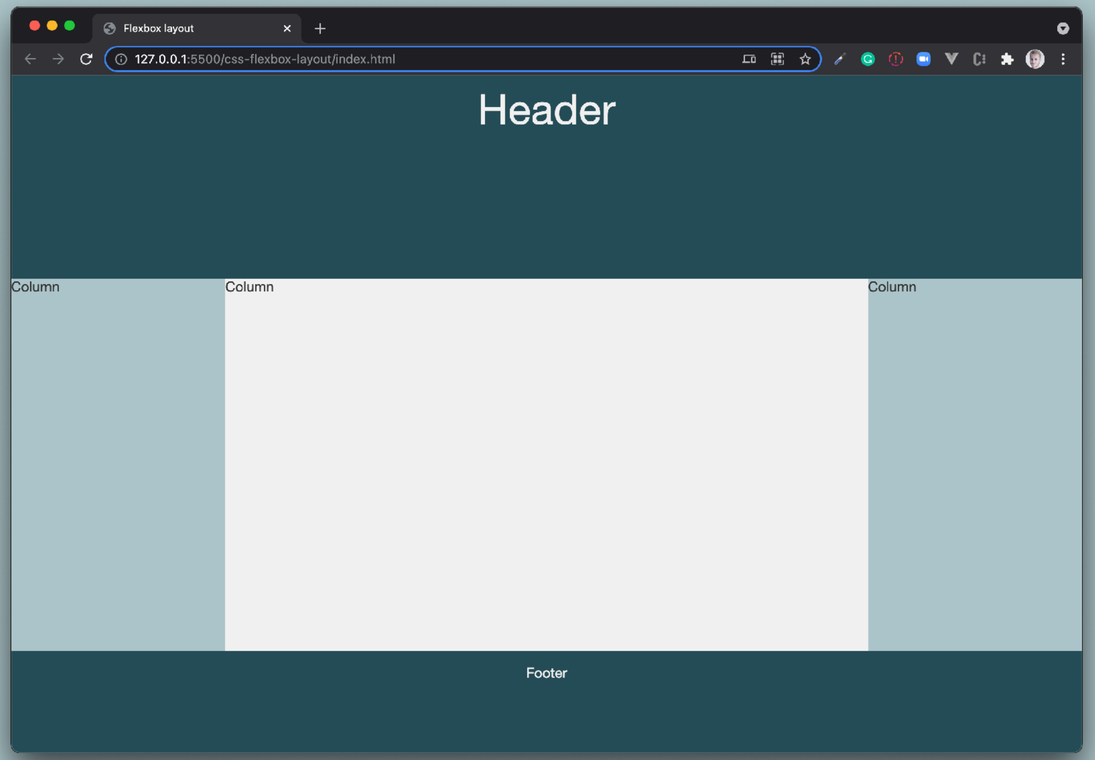
https://www.w3schools.com/whatis/whatis_htmldom.asp

CSSLAYOUT.IO

LAYOUTS AND PATTERNS MADE WITH CSS



css-grid-layout

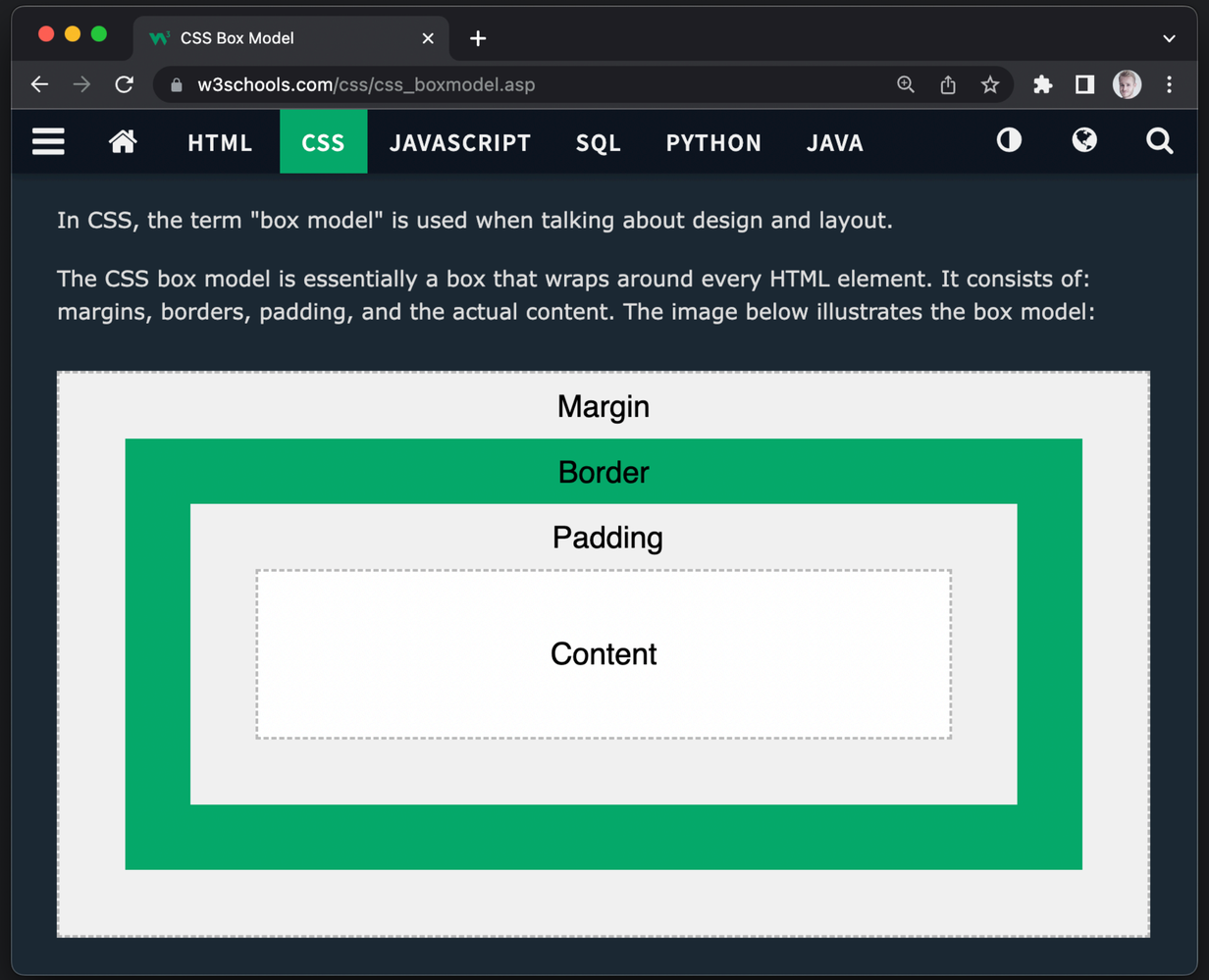


css-flexbox-layout

CSS templates

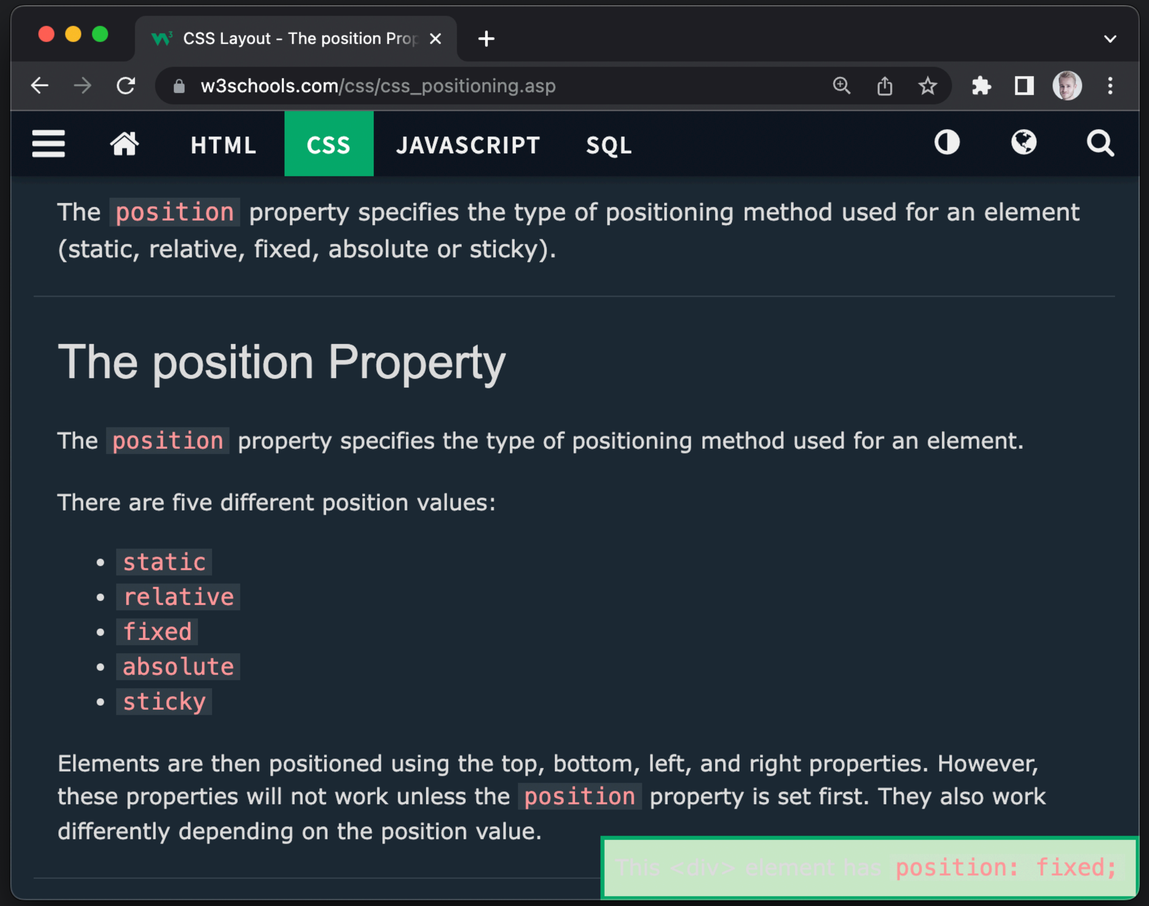
https://www.w3schools.com/css/css_templates.asp

The CSS Box Model



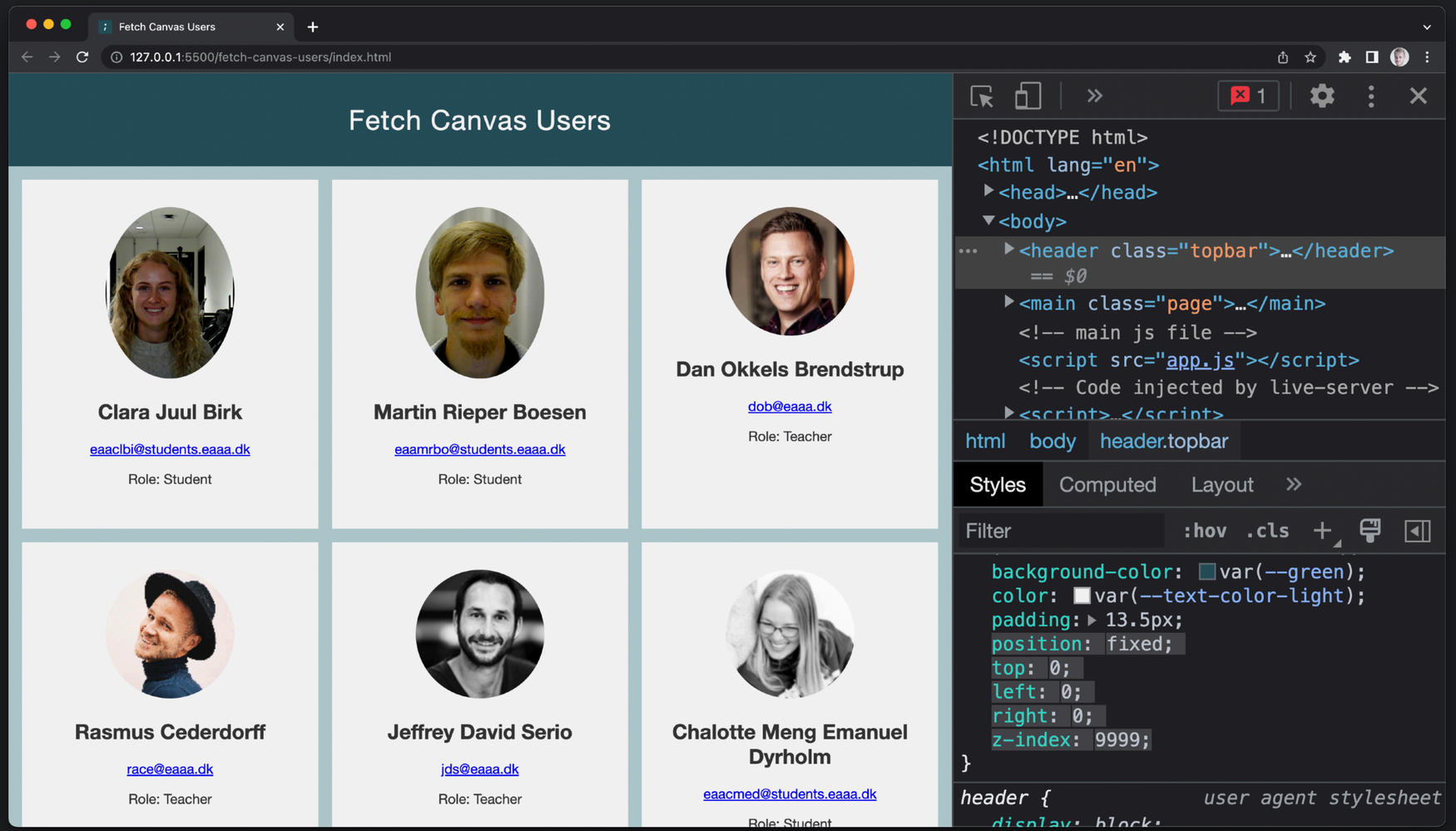
https://www.w3schools.com/css/css_boxmodel.asp

CSS Position



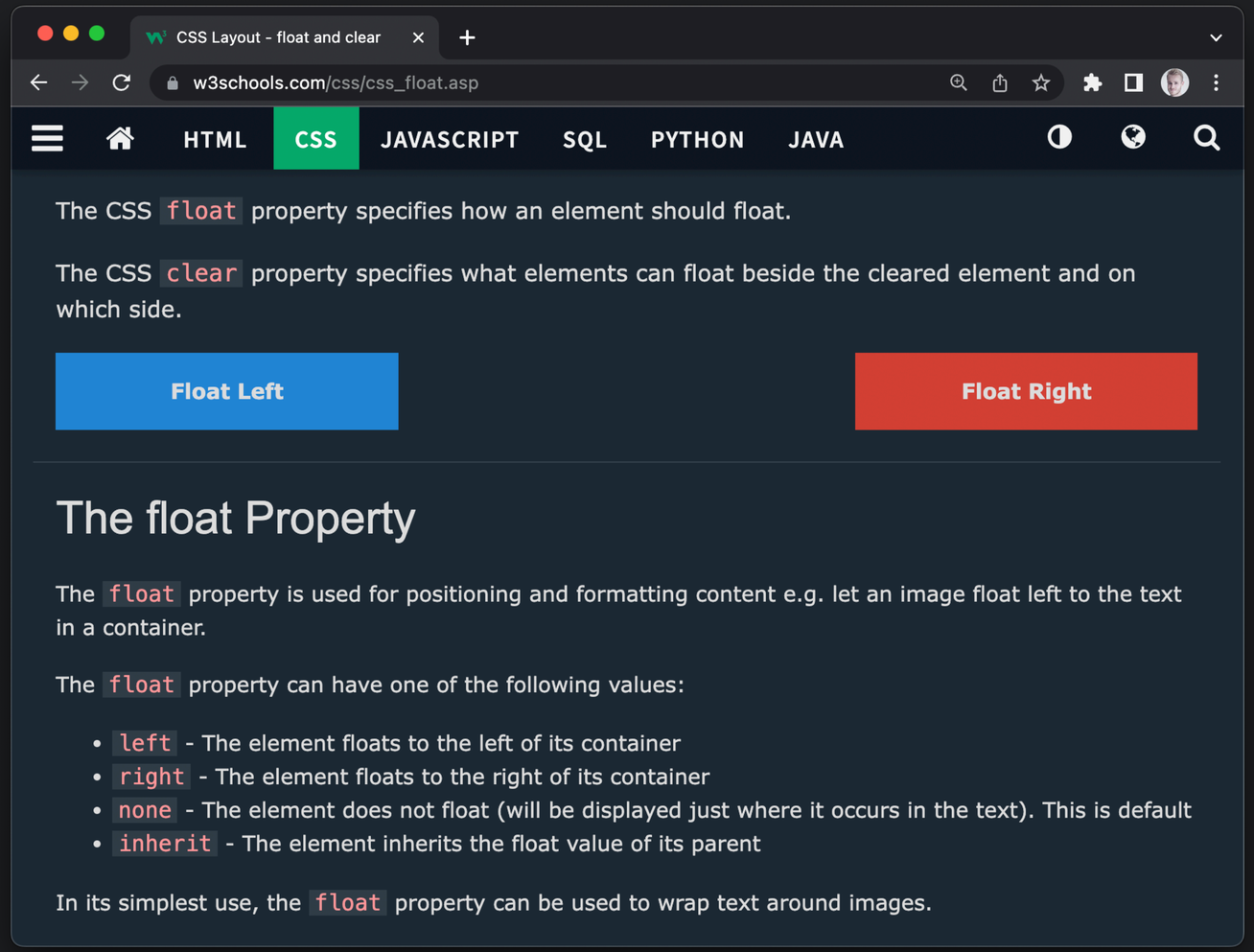
https://www.w3schools.com/css/css_positioning.asp

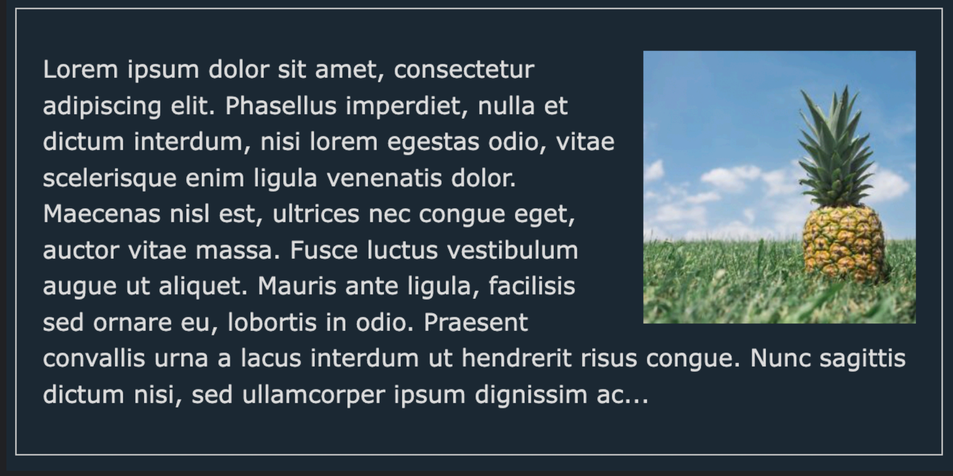
Fixed Header



https://www.w3schools.com/css/css_positioning.asp

CSS - Float & Clear

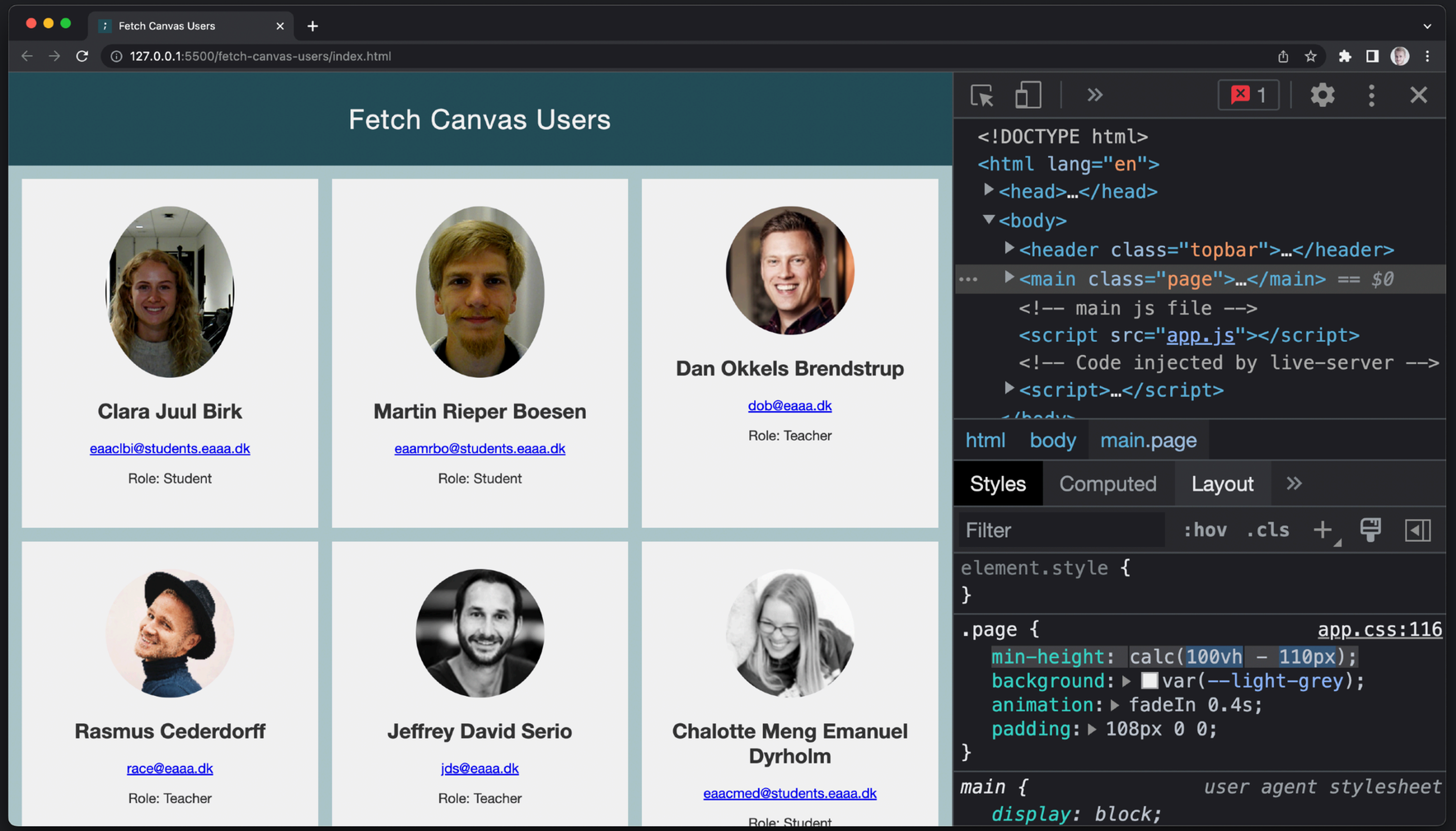





```css
img {
  float: right;
}
```

https://www.w3schools.com/css/css_float.asp

CSS calc



https://www.w3schools.com/cssref/func_calc.asp

CSS VARIABLES

`var()` - refer to a defined CSS variable.

CSS variables are declared inside the `:root` selector

CSS variables can be accessed through the entire CSS document.

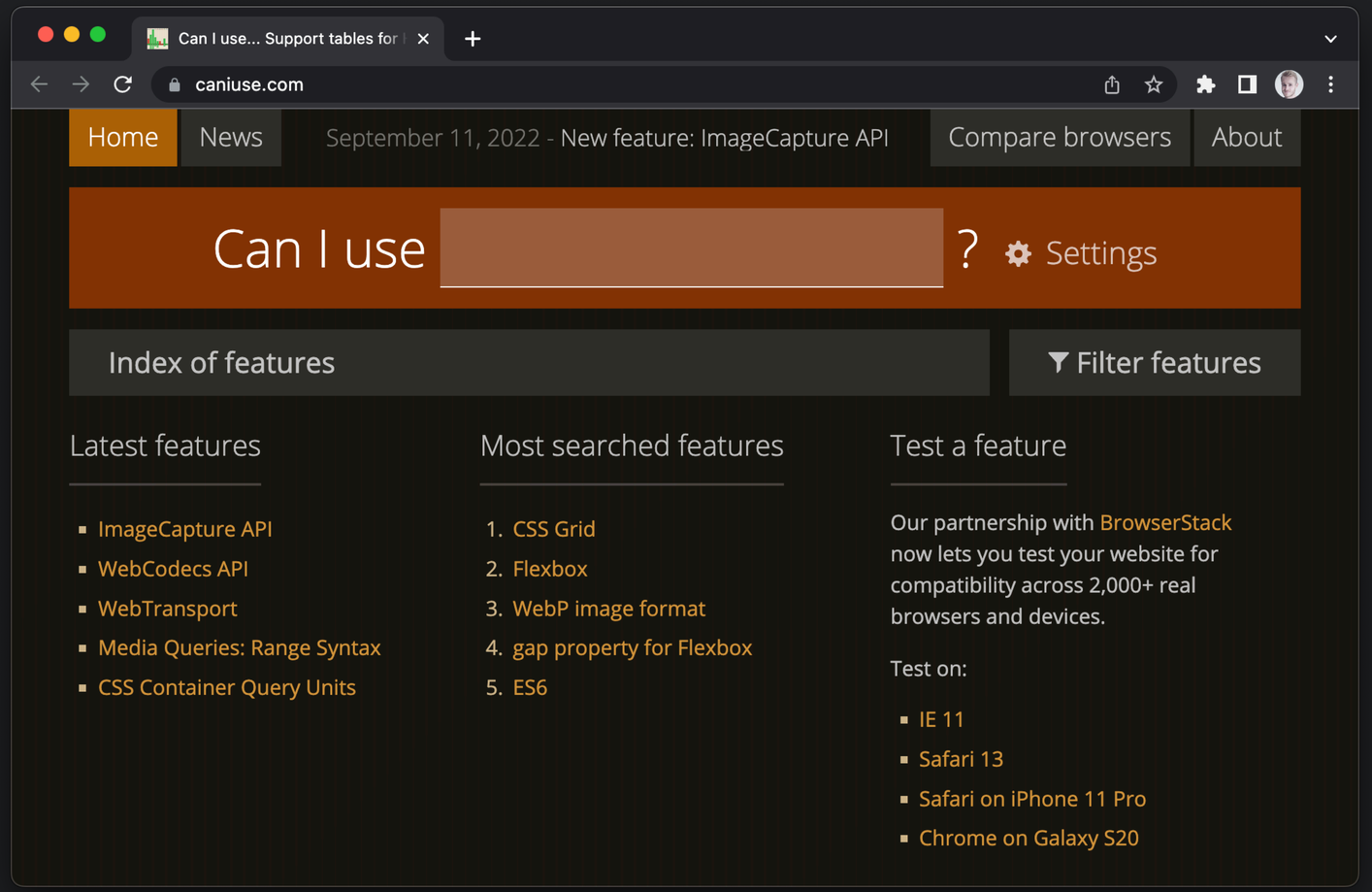
Makes the code easier to read (more understandable)

Makes it much easier to change the color values

https://www.w3schools.com/css/css3_variables.asp

```css
/* ---------- root variables ---------- */
:root {
  --green: rgb(38, 76, 89);
  --green-opacity: rgba(38, 76, 89, 0.2);
  --light-green: rgb(172, 198, 201);
  --light-grey: #f1f1f4;
  --text-color-light: #f1f1f1;
  --text-color-dark: #333;
  --white: #fff;
  --font-family: "Helvetica Neue", Helvetica, Arial, sans-serif;
}

/* ---------- styling ---------- */
html,
body {
  color: var(--text-color-dark);
  font-family: var(--font-family);
  margin: 0;
  padding: 0;
  border: 0;
  outline: 0;
  background-color: var(--light-grey);
}

a {
  color: var(--green);
}

.grid > article {
  text-align: center;
  padding: 2em 1em;
  background-color: var(--light-green);
  transition: 0.3s;
}
```

Compatibility: Can I use?



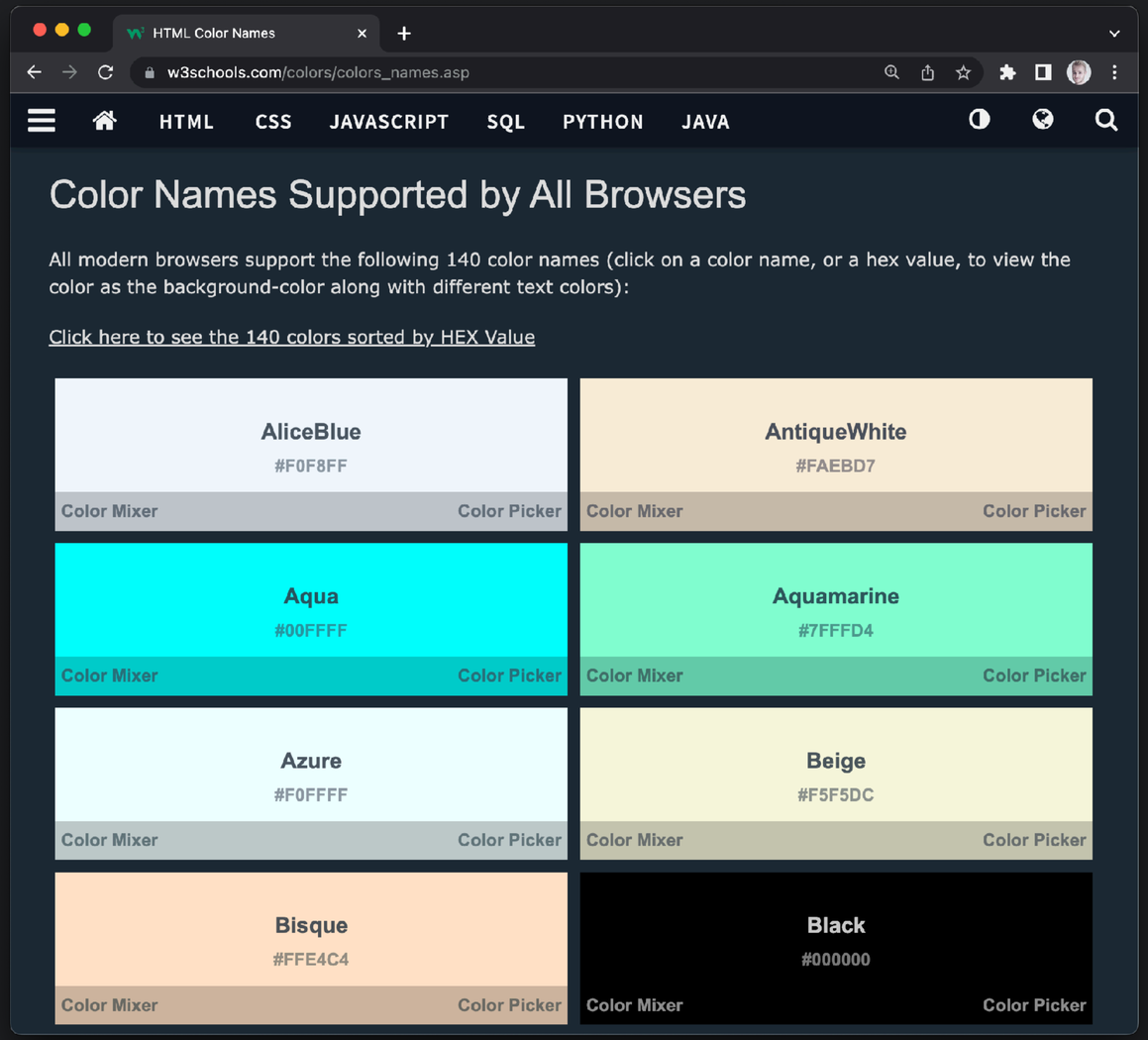
https://caniuse.com/

Further CSS

Overflow: https://www.w3schools.com/css/css_overflow.asp

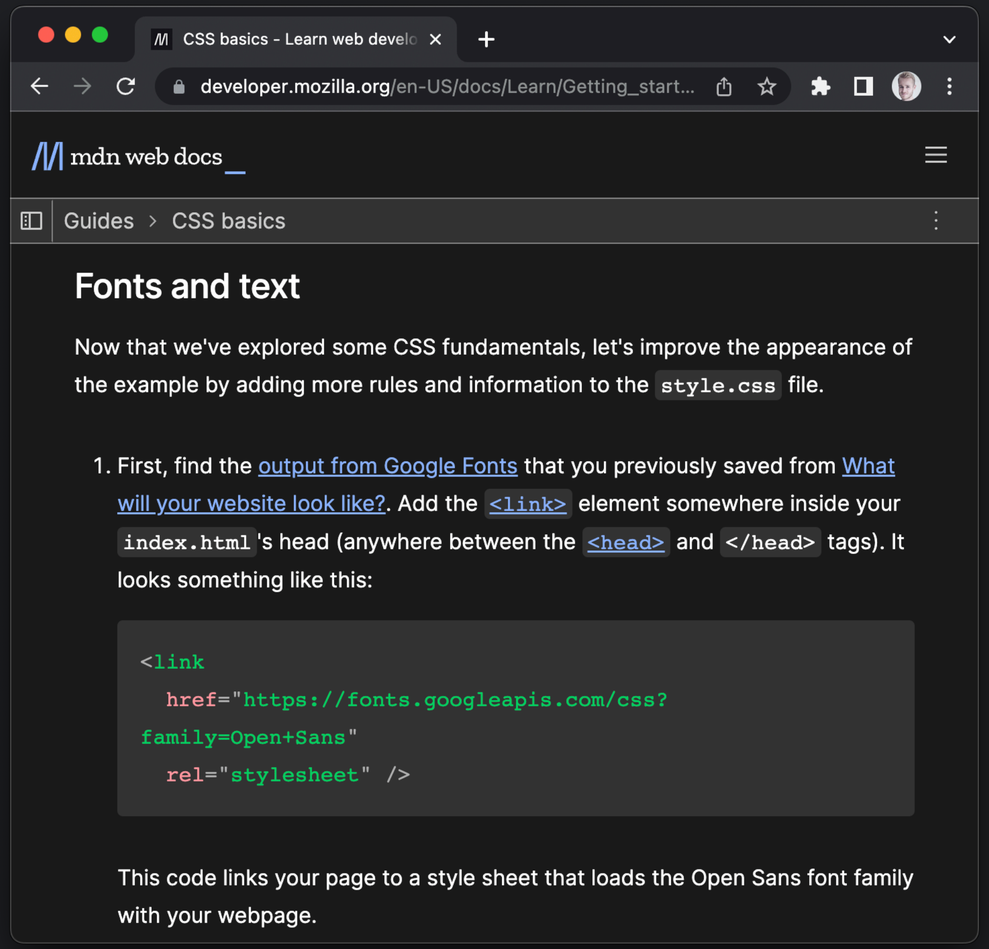
CSS Tutorial: https://www.w3schools.com/css/default.asp

Web Safe Colors



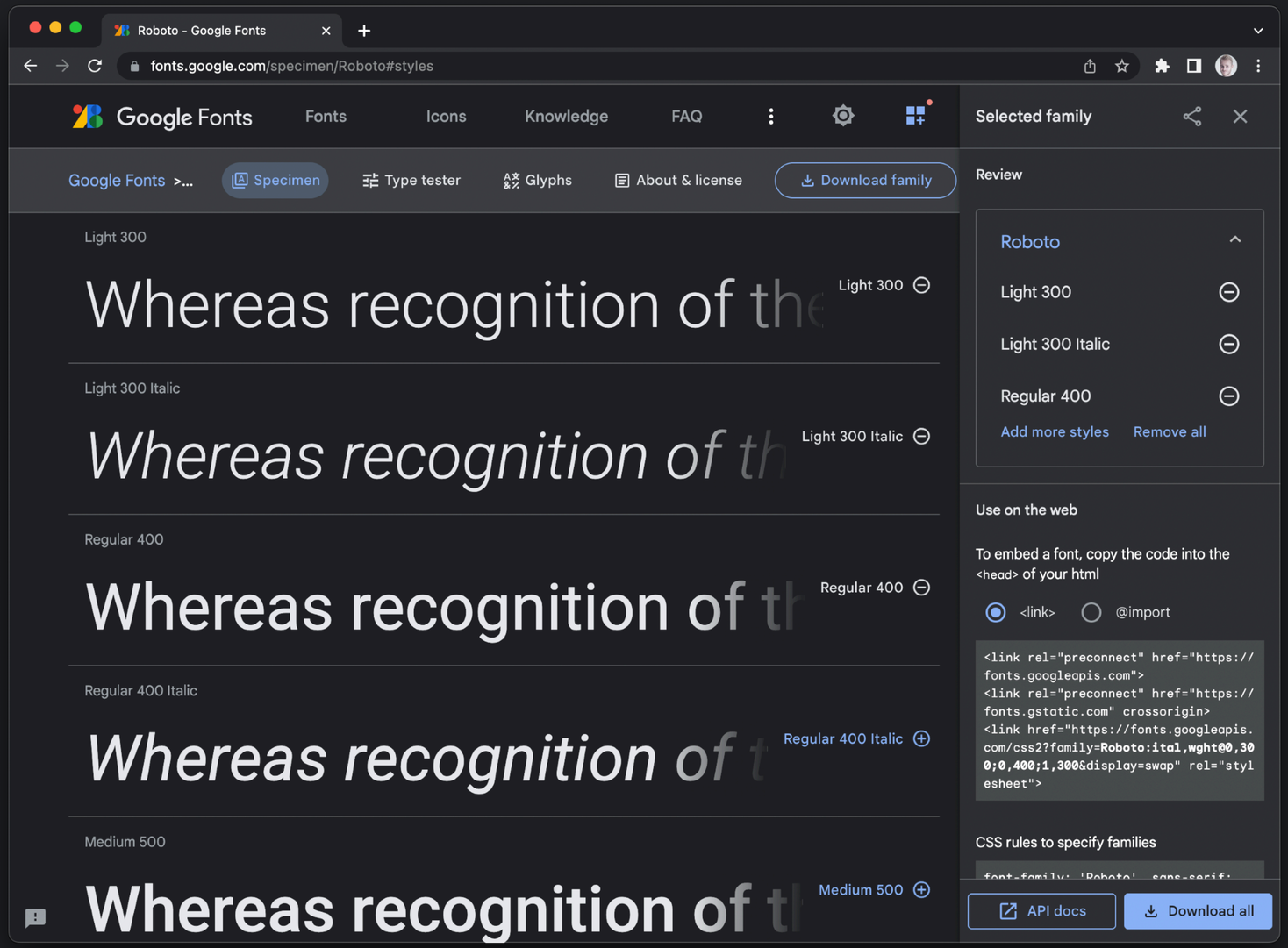
https://www.w3schools.com/colors/colors_names.asp

Fonts



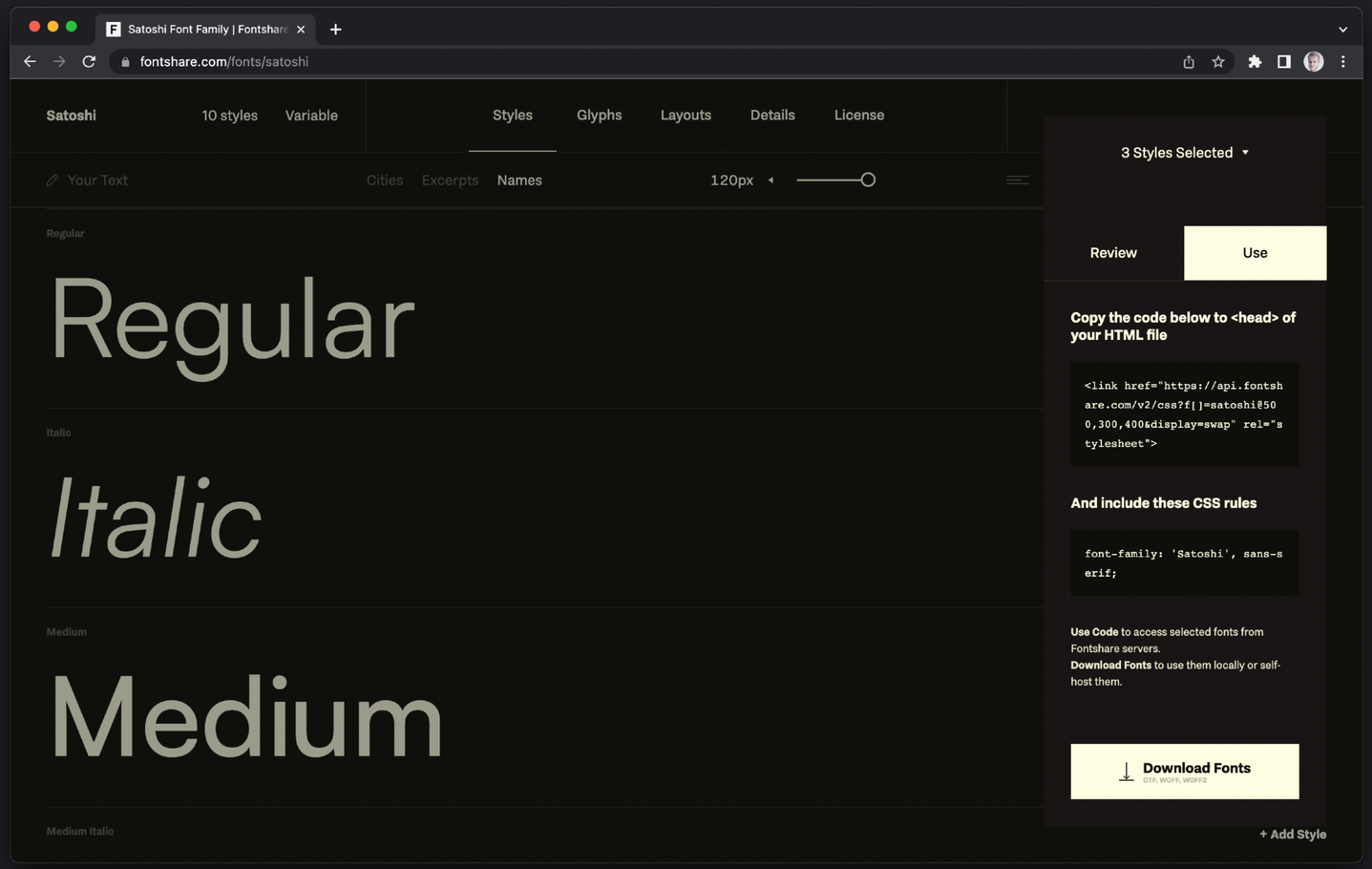
- Read: Fonts and text
- Use “Open Sans” in your solution.
- Types:
  - Web Safe: universally installed across all browsers and devices.
  - Font Files (ex. cederdorff.com). @font-face rule
  - External Fonts (scripts): `link` or `import` from external font resources like Google Font and Fontshare

Fonts



- Explore fonts.google.com
- Select fonts and use in your CSS.

Fonts



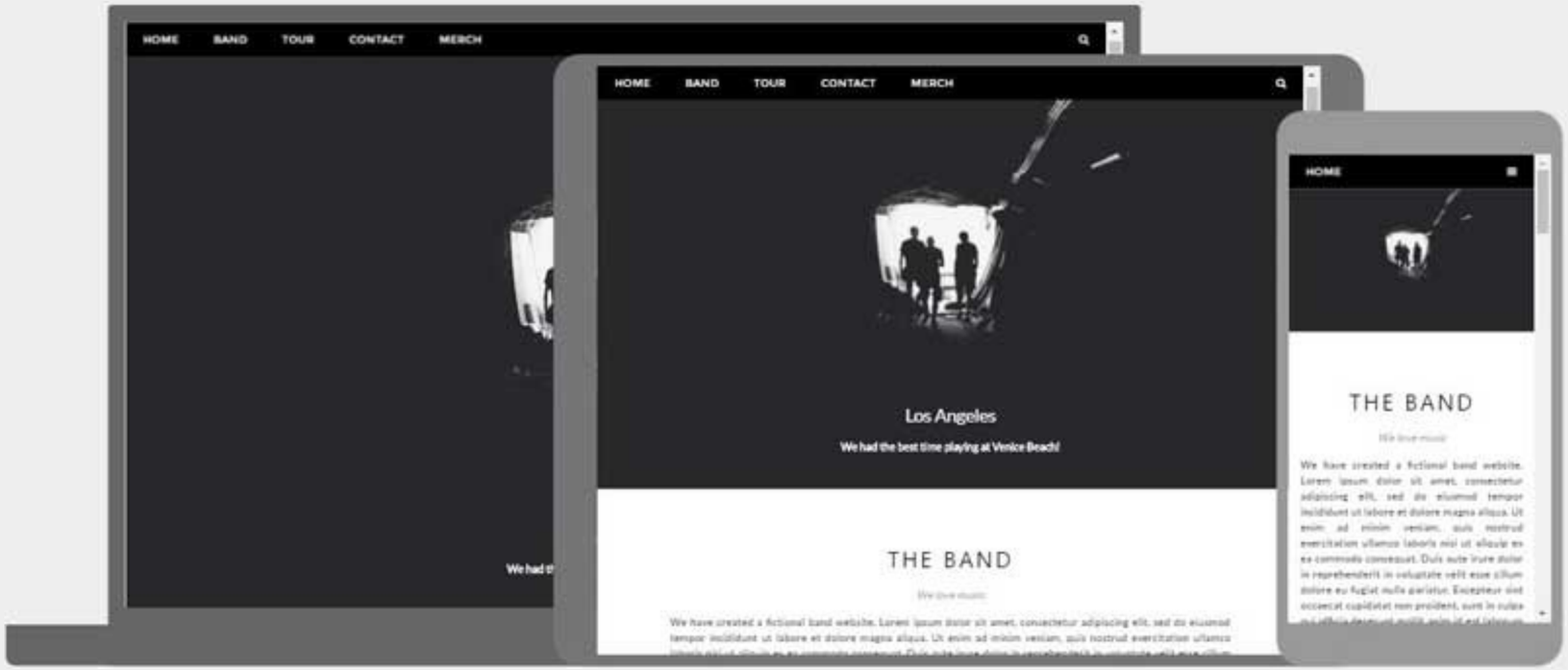
- Google Fonts alternative: fontshare.com
- Select fonts and use in your CSS.

Responsive Web Design

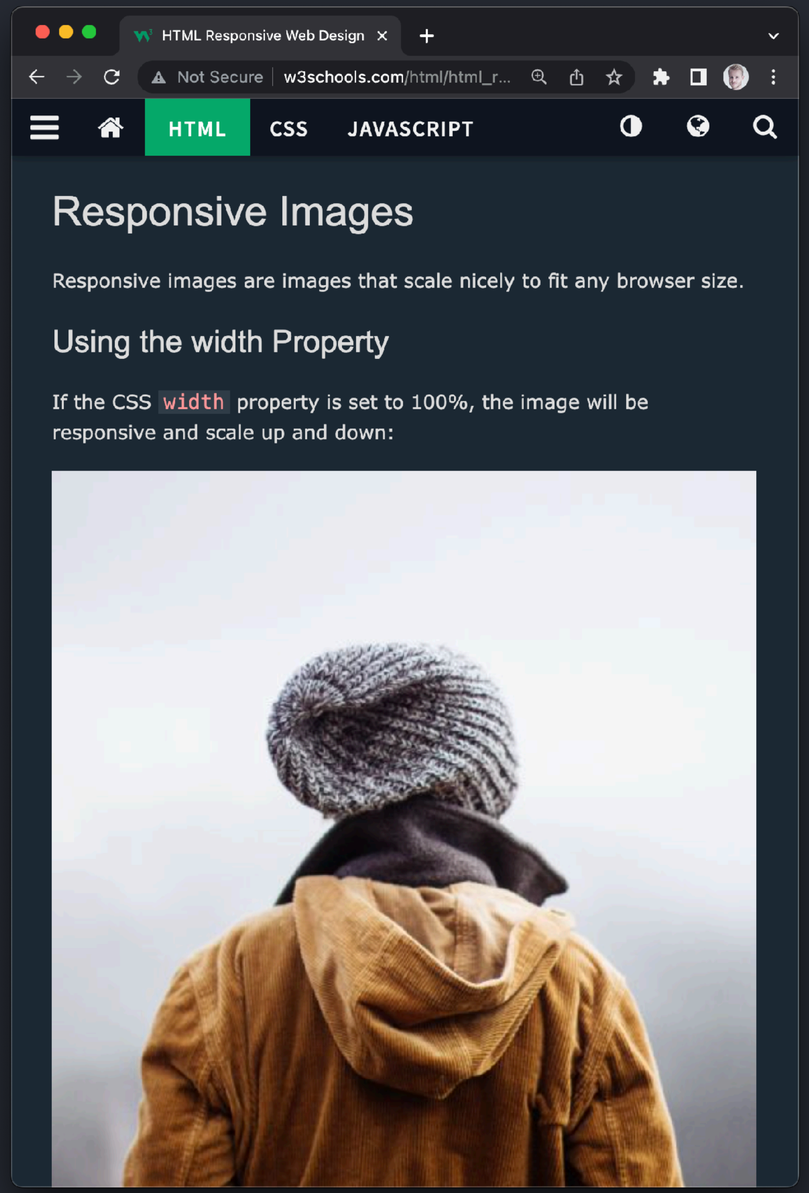
- Create Web Design that looks good across all devices!
- Automatically adjust for different screen sizes and viewports.
- Resize, hide, shrink, or enlarge.

https://www.w3schools.com/html/html_responsive.asp

Resize, hide, shrink, enlarge



https://www.w3schools.com/w3css/tryw3css_templates_band.htm
All W3School Templates



Responsive Images

```
<img src="img/cookies.jpg" alt="Arla Cookies" />
```

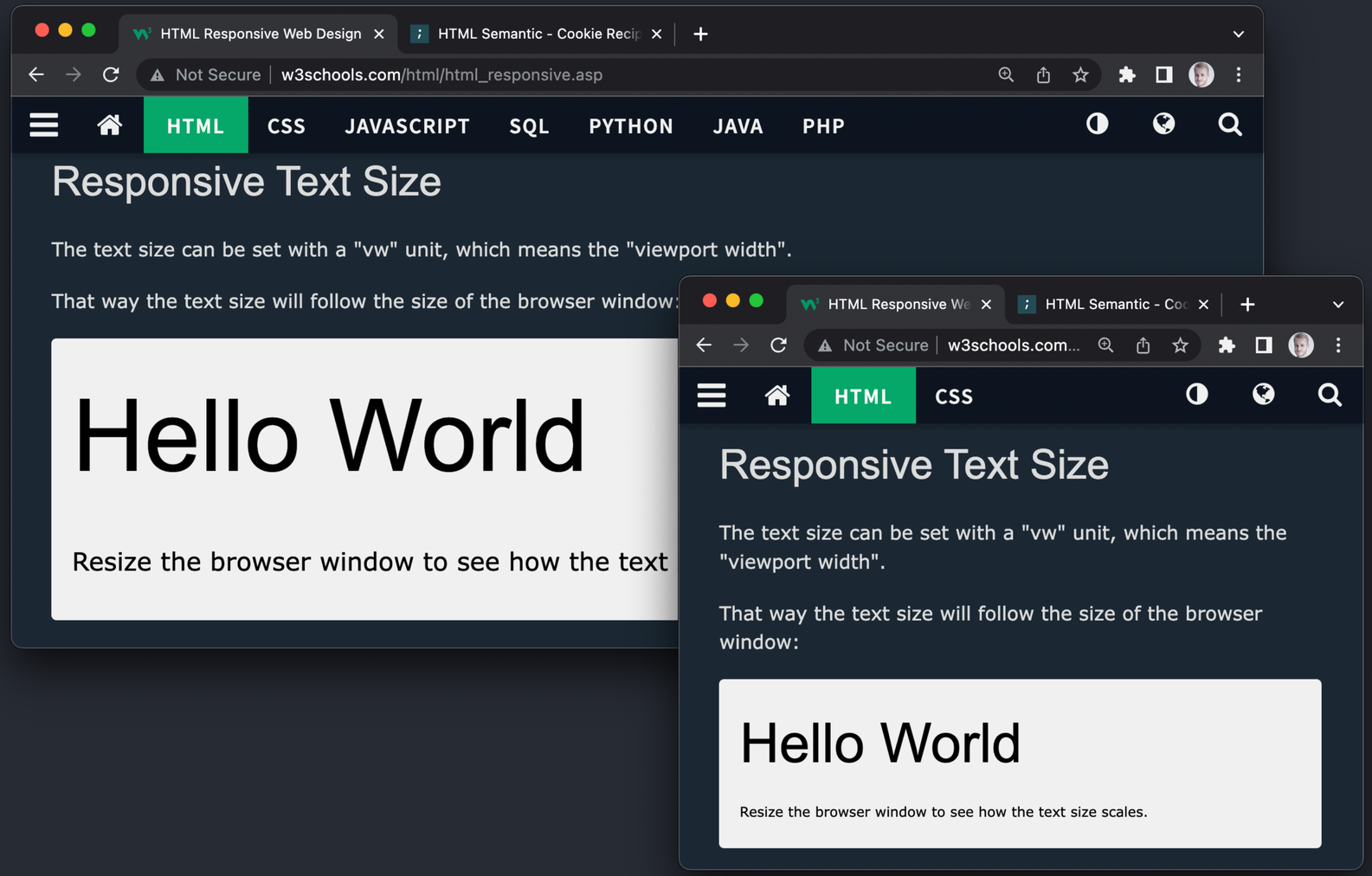
```
img {
  max-width: 100%;
  height: auto;
}
```

https://www.w3schools.com/css/css_rwd_images.asp

Responsive Text Size

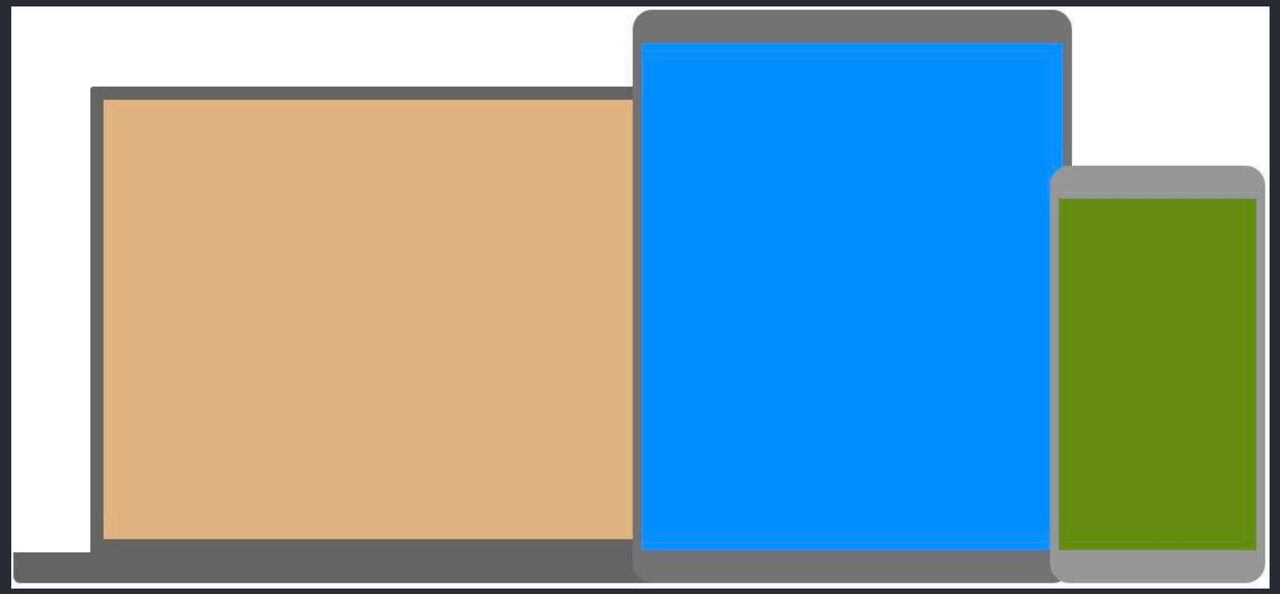
```
<h1>Cookies</h1>
```

```
h1 {
  font-size: 10vw;
}
```



https://www.w3schools.com/html/html_responsive.asp

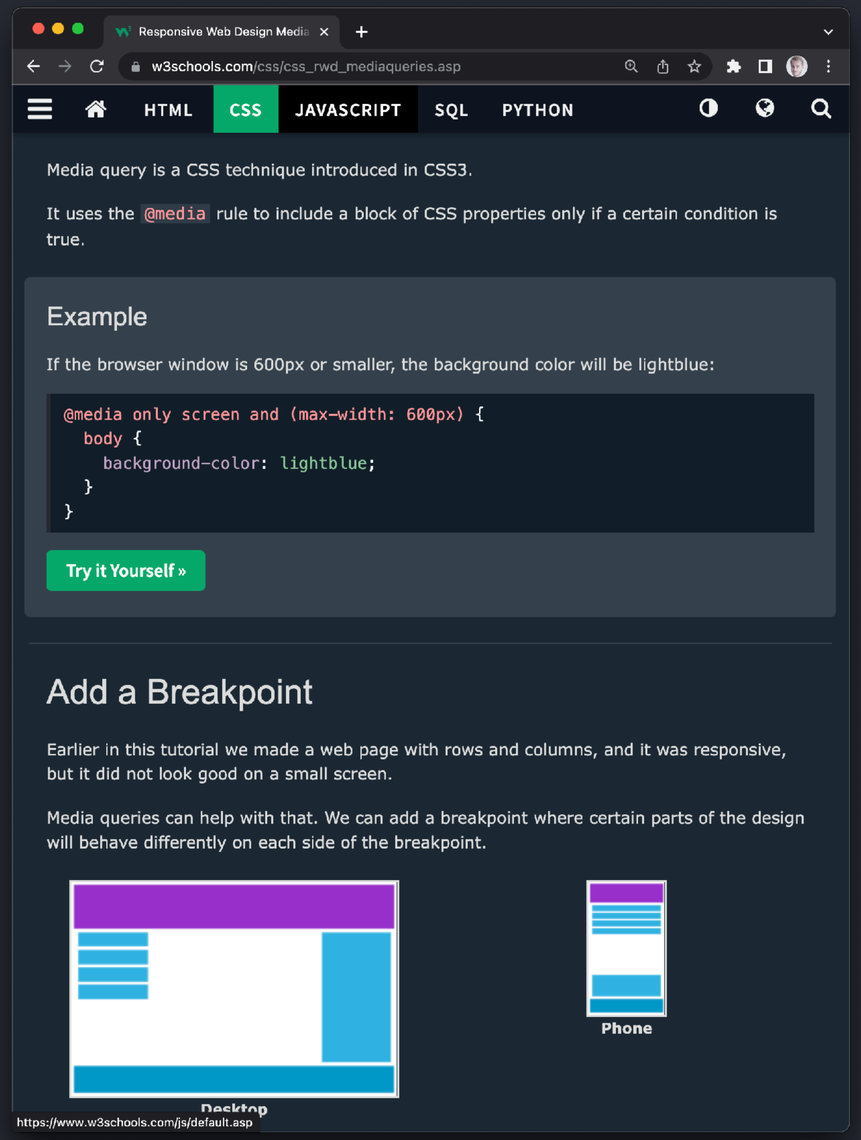
Media Queries



```css
/* Set the background color of body to tan */
body {
  background-color: tan;
}

/* On screens that are 992px or less, set the background color to blue */
@media screen and (max-width: 992px) {
  body {
    background-color: blue;
  }
}

/* On screens that are 600px or less, set the background color to olive */
@media screen and (max-width: 600px) {
  body {
    background-color: olive;
  }
}
```

https://www.w3schools.com/css/css3_mediaqueries_ex.asp

Media Queries




```
<h1>Cookies</h1>
```

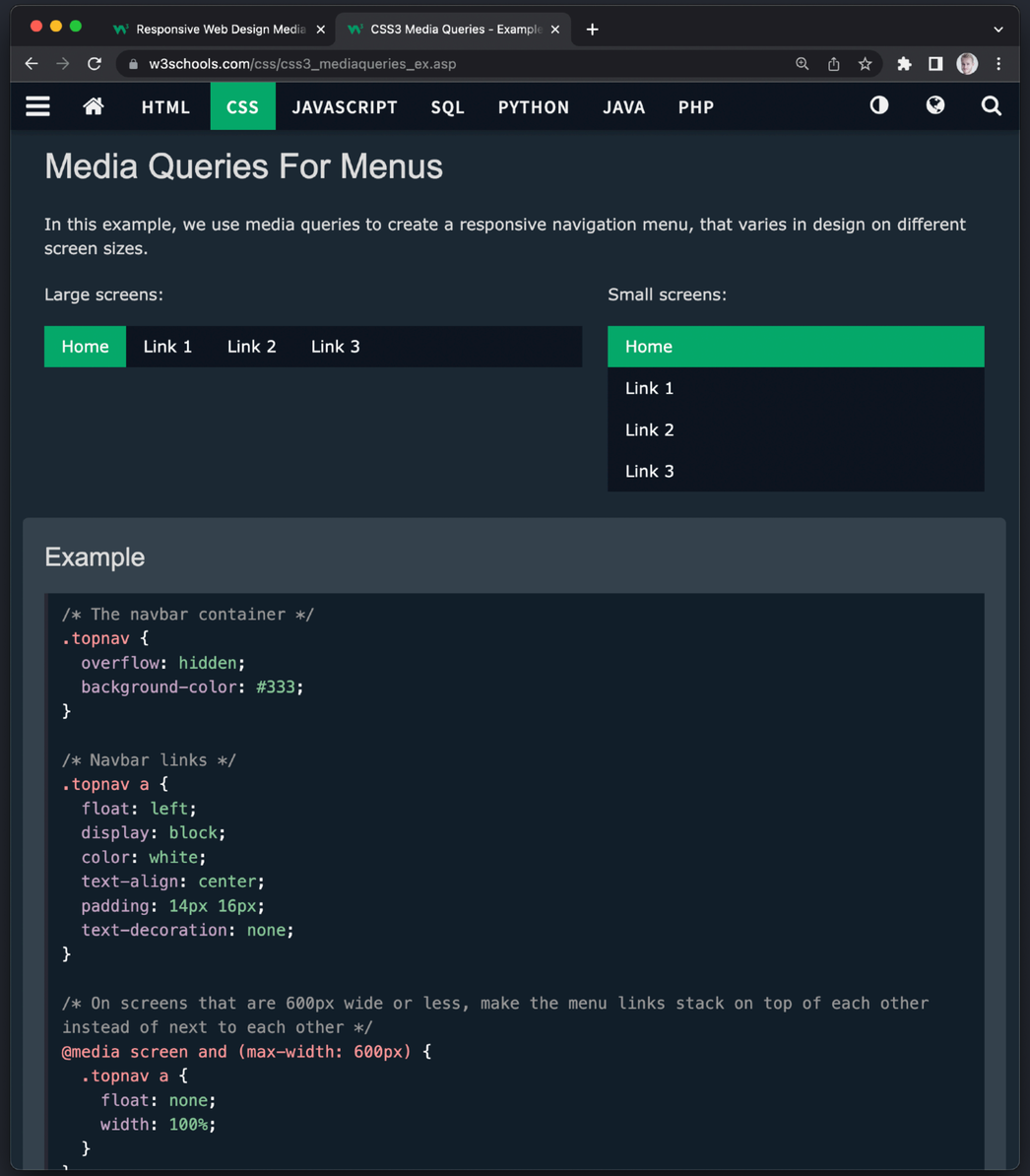
```
.header h1 {
    font-size: 1.3em;
    margin-top: 0;
    margin: 10px 0 0;
}

@media (min-width: 768px) {
    .header h1 {
        font-size: 1.5em;
        margin: 0;
    }
}

@media (min-width: 992px) {
    .header h1 {
        font-size: 1.8em;
    }
}
```

https://www.w3schools.com/css/css_rwd_mediaqueries.asp

Design for Mobile First

Start with mobile, then add CSS for tablet, desktop and other screen sizes.

https://www.w3schools.com/css/css_rwd_mediaqueries.asp

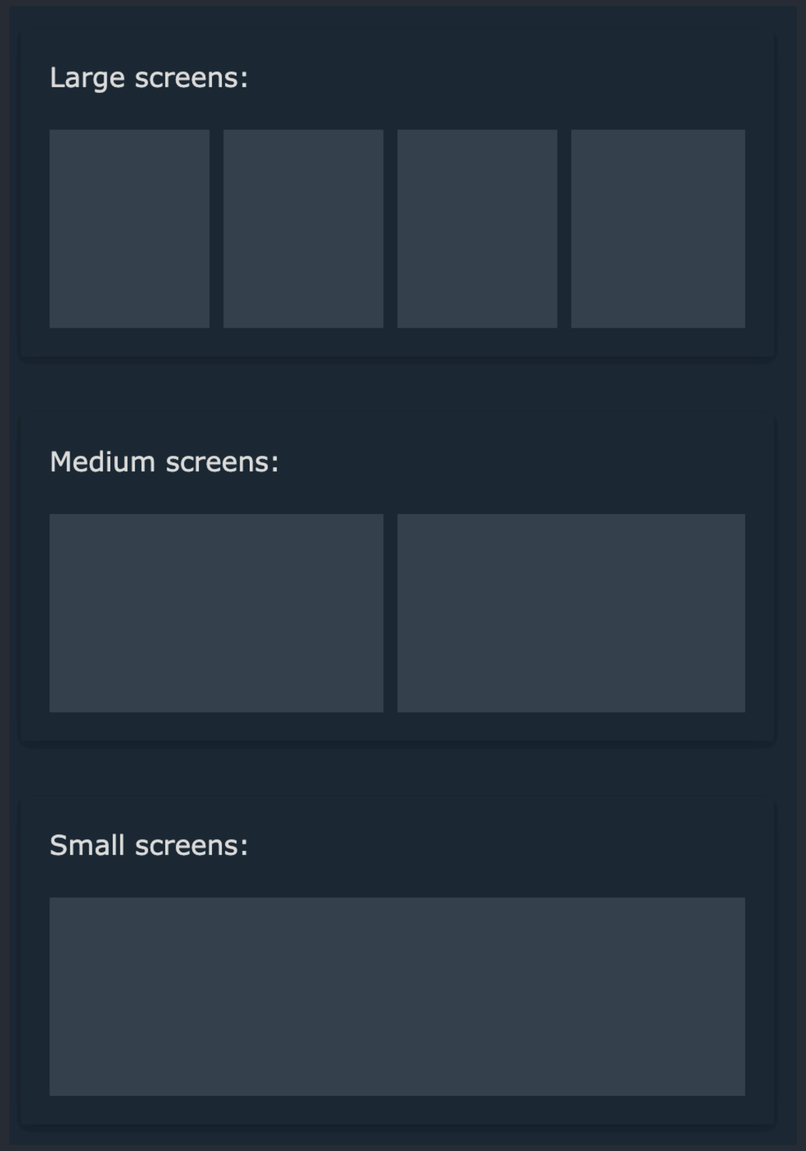


Media Queries & Menus

```css
/* The navbar container */
.topnav {
  overflow: hidden;
  background-color: #333;
}

/* Navbar links */
.topnav a {
  float: left;
  display: block;
  color: white;
  text-align: center;
  padding: 14px 16px;
  text-decoration: none;
}

/* On screens that are 600px wide or less, make the menu links stack on top o
@media screen and (max-width: 600px) {
  .topnav a {
    float: none;
    width: 100%;
  }
}
```

https://www.w3schools.com/css/css3_mediaqueries_ex.asp

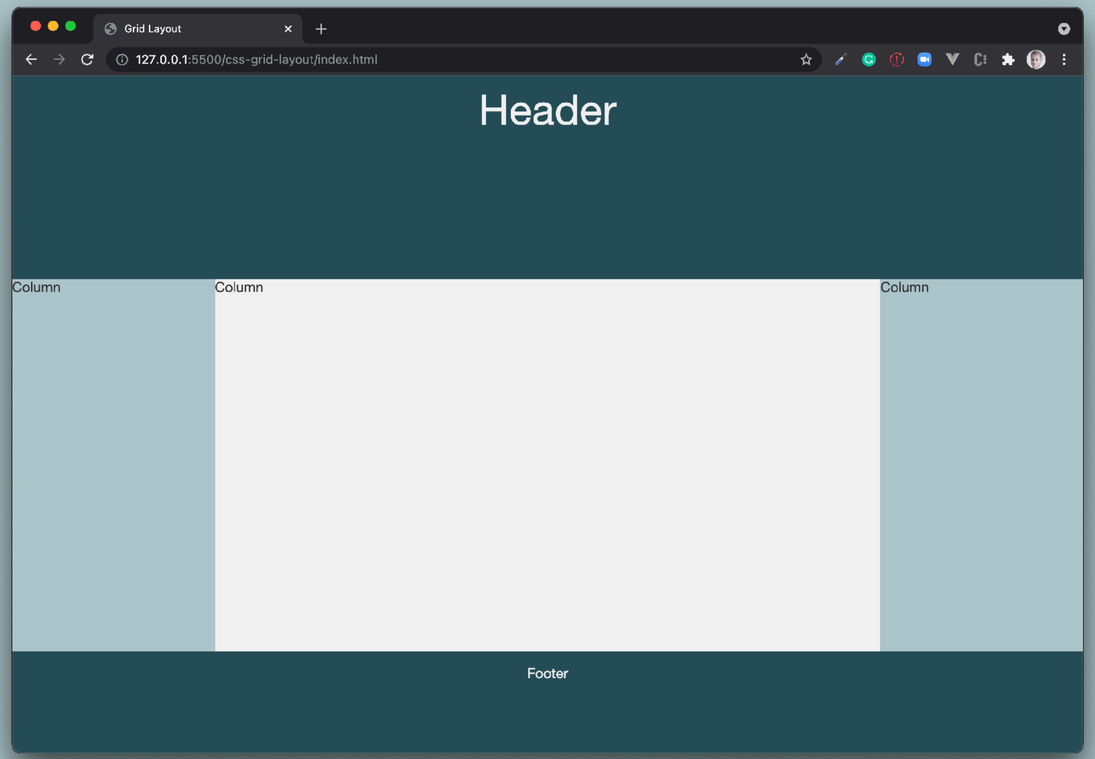
Media Queries For Columns



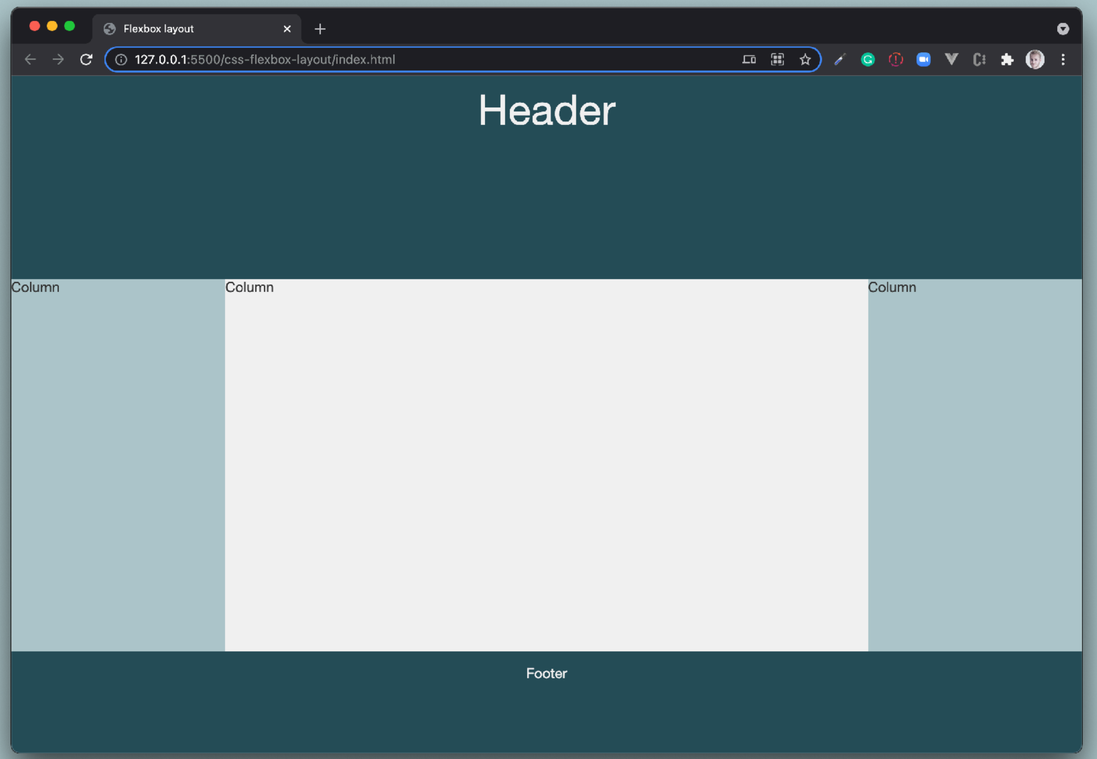

```css
/* Create four equal columns that floats next to each other */
.column {
  float: left;
  width: 25%;
}

/* On screens that are 992px wide or less, go from four columns to two columns */
@media screen and (max-width: 992px) {
  .column {
    width: 50%;
  }
}

/* On screens that are 600px wide or less, make the columns stack on top of each other instea
@media screen and (max-width: 600px) {
  .column {
    width: 100%;
  }
}
```

https://www.w3schools.com/css/css3_mediaqueries_ex.asp

Grids



css-grid-layout



css-flexbox-layout

Responsive Web Design Templates

https://www.w3schools.com/css/css_rwd_templates.asp

Hvordan er “formen”?

Tak for i dag

Webudvikling

kea
KØBENHAVNS ERHVERVSAKADEMI