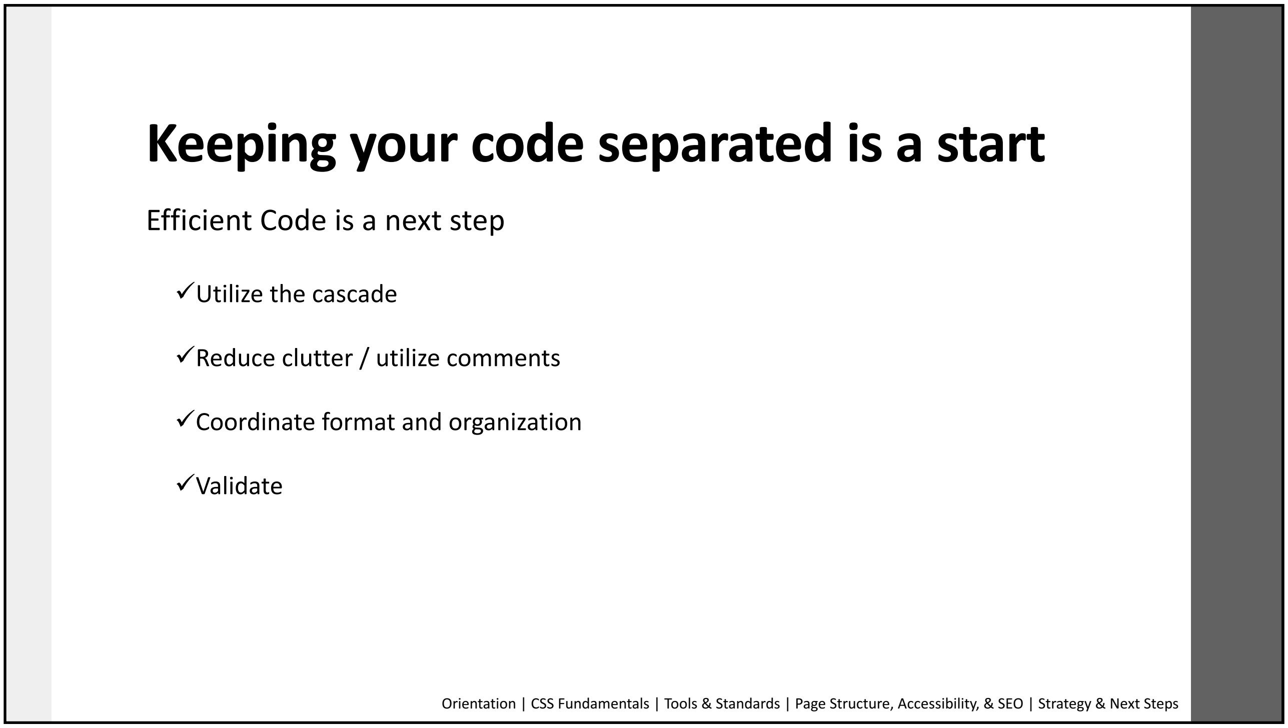
Front End Web

CSS & HTML Best Practices
Review and Discussion

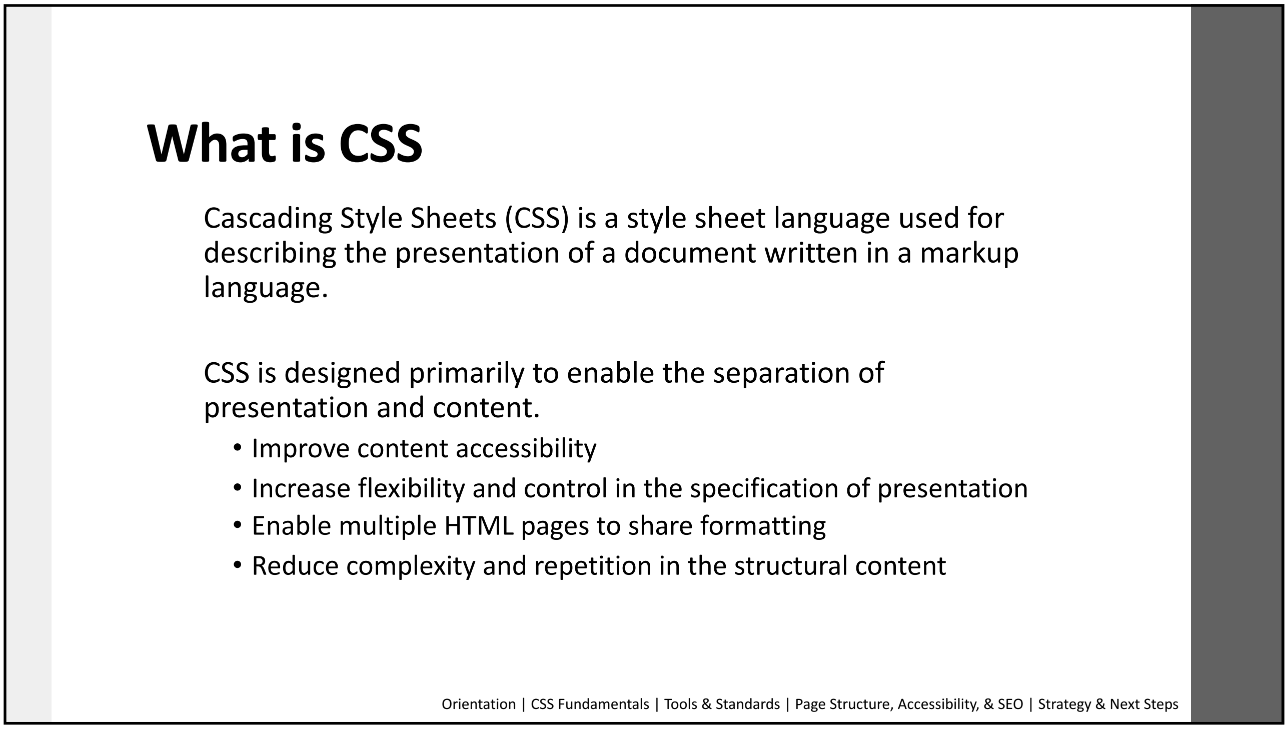
Separating structural, style, and interactive makes your life easier



Keeping your code separated is a start

Efficient Code is a next step

- ✓Utilize the cascade
- ✓Reduce clutter / utilize comments
- ✓Coordinate format and organization
- ✓Validate



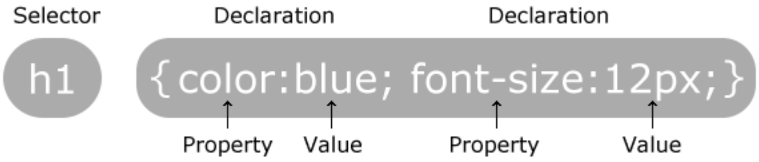
What is CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language.

CSS is designed primarily to enable the separation of presentation and content.

- Improve content accessibility
- Increase flexibility and control in the specification of presentation
- Enable multiple HTML pages to share formatting
- Reduce complexity and repetition in the structural content

What is CSS



Selector

Declaration

Declaration

```
h1 {color:blue; font-size:12px;}
```

Property

Value

Property

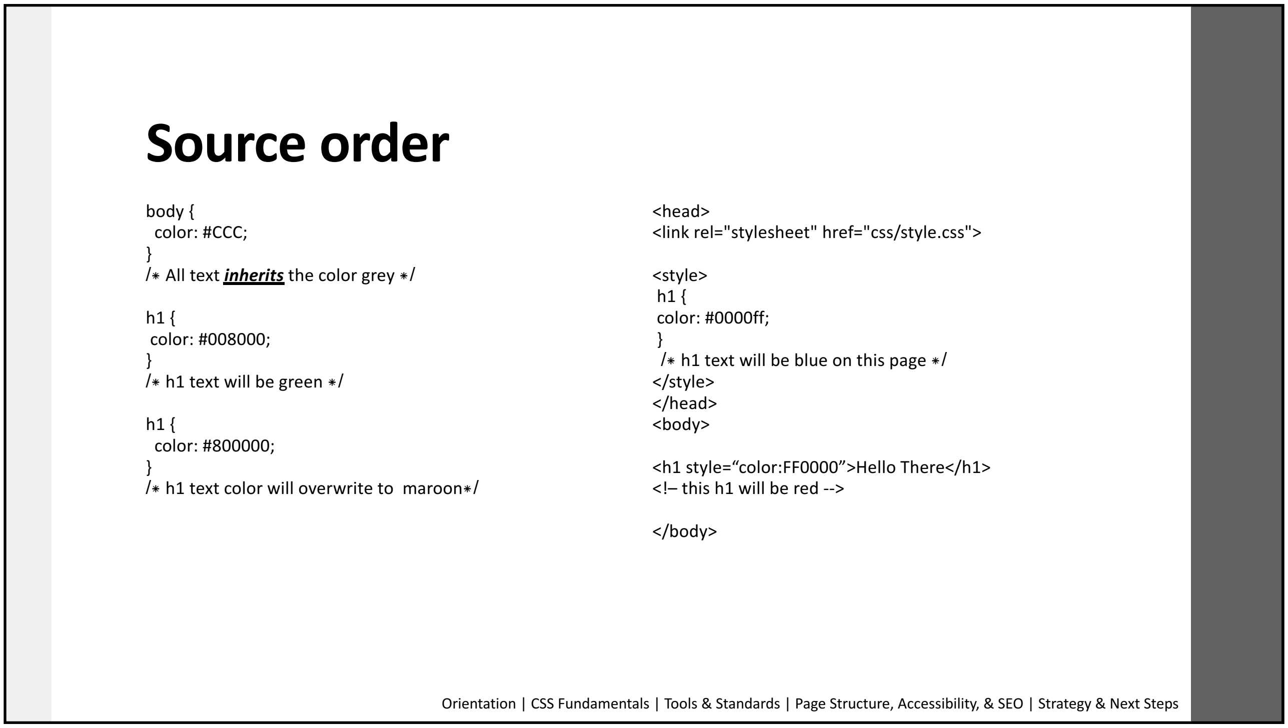
Value

Cascade

CSS is an acronym of *Cascading Style Sheets*, which indicates that the notion of the cascade is important. At its most basic level it indicates that the order of CSS rules matter, but it's more complex than that. What selectors win out in the cascade depends on three factors (these are listed in order of weight — earlier ones will overrule later ones):

- Source order
- Importance
- Specificity

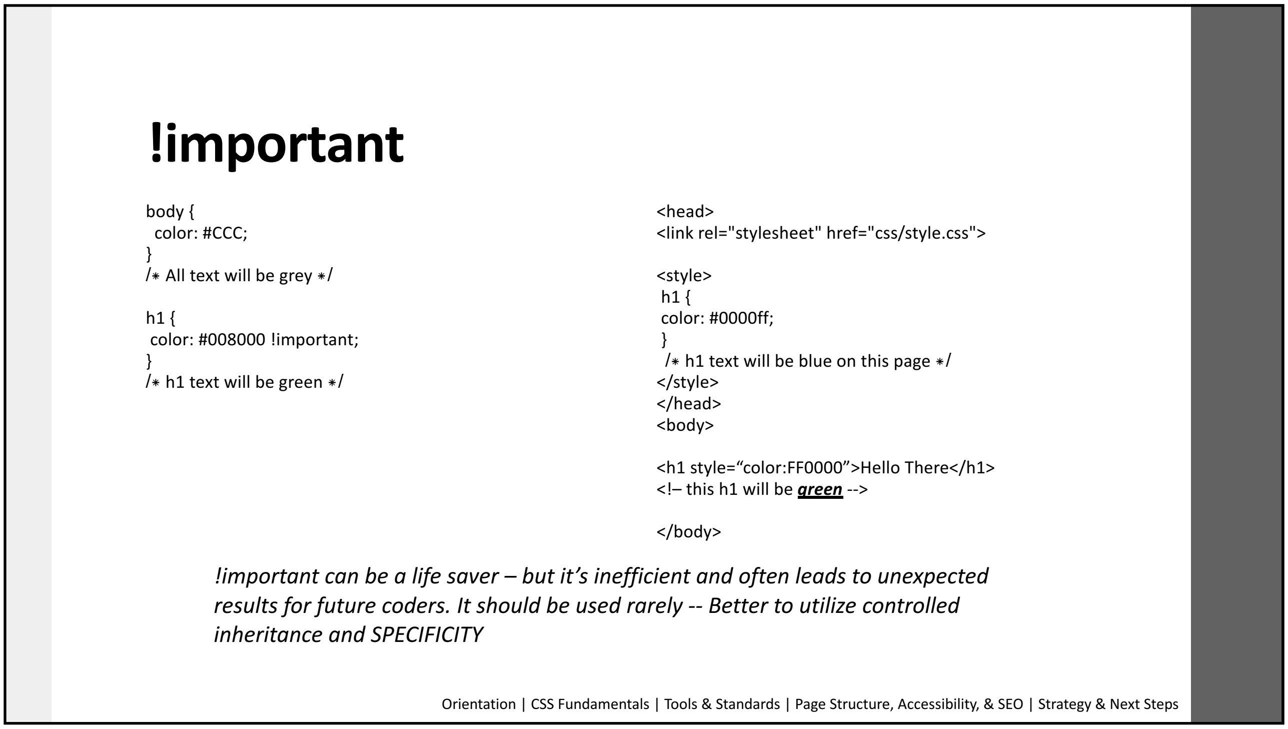
Source: https://en.wikipedia.org/wiki/Cascading_Style_Sheets#Selector



Source order

```
body {
  color: #CCC;
}
/∗ All text inherits the color grey ∗/

h1 {
 color: #008000;
}
/∗ h1 text will be green ∗/

h1 {
  color: #800000;
}
/∗ h1 text color will overwrite to  maroon∗/
```

```
<head>
<link rel="stylesheet" href="css/style.css">

<style>
 h1 {
 color: #0000ff;
 }
  /∗ h1 text will be blue on this page ∗/
</style>
</head>
<body>

<h1 style=“color:FF0000”>Hello There</h1>
<!– this h1 will be red -->

</body>
```

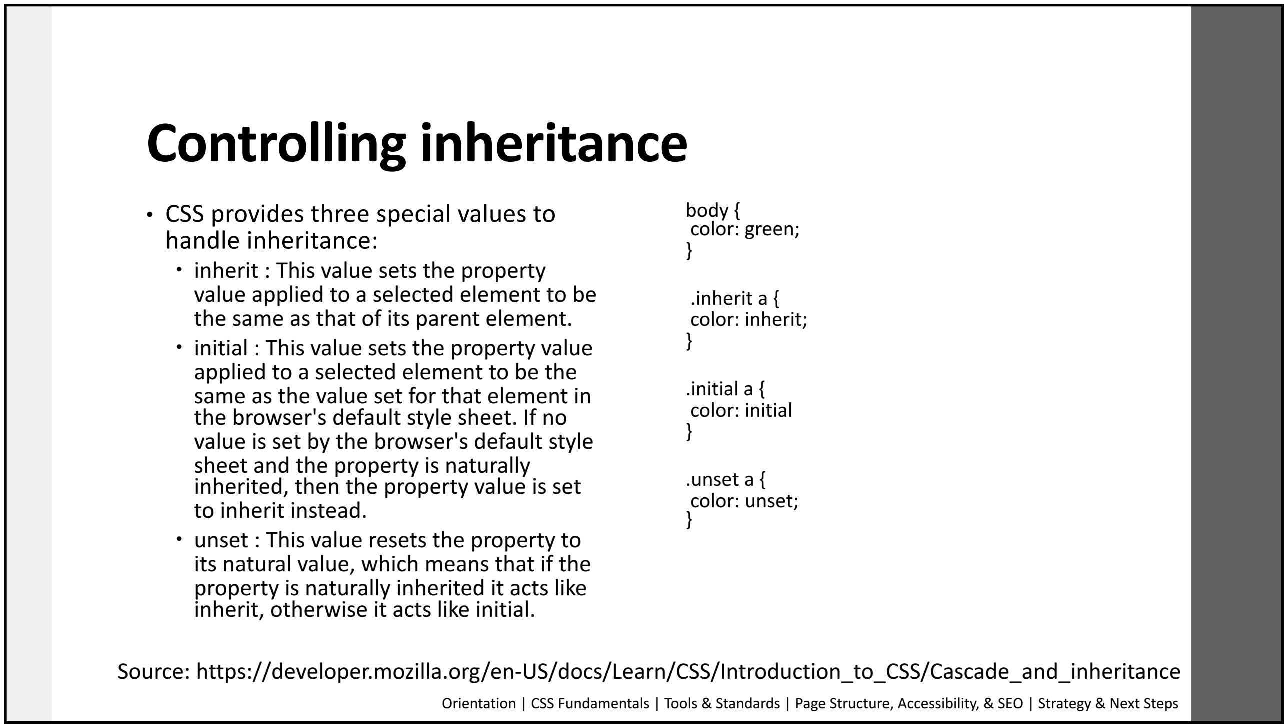


!important

```
body {
  color: #CCC;
}
/∗ All text will be grey ∗/

h1 {
 color: #008000 !important;
}
/∗ h1 text will be green ∗/
```

```
<head>
<link rel="stylesheet" href="css/style.css">

<style>
 h1 {
 color: #0000ff;
 }
  /∗ h1 text will be blue on this page ∗/
</style>
</head>
<body>

<h1 style=“color:FF0000”>Hello There</h1>
<!– this h1 will be green -->

</body>
```

!important can be a life saver – but it’s inefficient and often leads to unexpected results for future coders. It should be used rarely -- Better to utilize controlled inheritance and SPECIFICITY

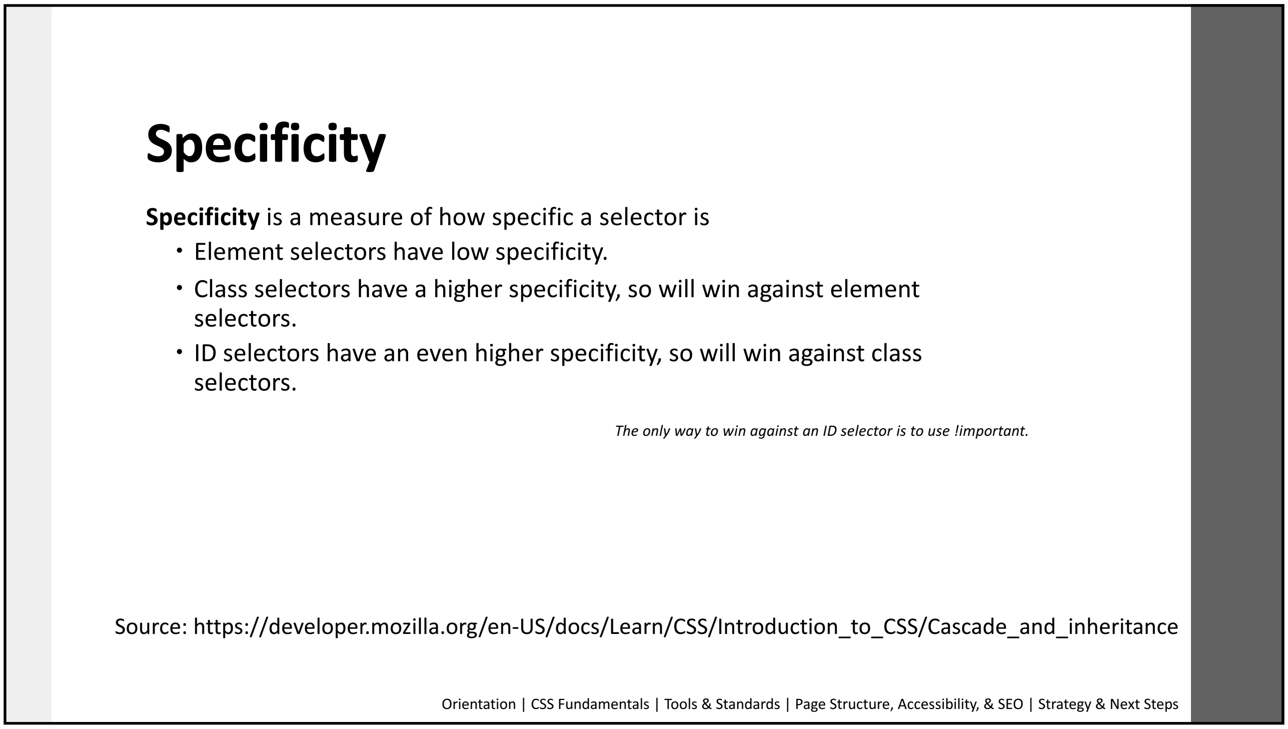


Controlling inheritance

- CSS provides three special values to handle inheritance:
  - inherit : This value sets the property value applied to a selected element to be the same as that of its parent element.
  - initial : This value sets the property value applied to a selected element to be the same as the value set for that element in the browser's default style sheet. If no value is set by the browser's default style sheet and the property is naturally inherited, then the property value is set to inherit instead.
  - unset : This value resets the property to its natural value, which means that if the property is naturally inherited it acts like inherit, otherwise it acts like initial.

```
body {
 color: green;
}

 .inherit a {
 color: inherit;
}

.initial a {
 color: initial
}

.unset a {
 color: unset;
}
```

Source: https://developer.mozilla.org/en-US/docs/Learn/CSS/Introduction_to_CSS/Cascade_and_inheritance



Specificity

Specificity is a measure of how specific a selector is

- Element selectors have low specificity.
- Class selectors have a higher specificity, so will win against element selectors.
- ID selectors have an even higher specificity, so will win against class selectors.

The only way to win against an ID selector is to use !important.

Source: https://developer.mozilla.org/en-US/docs/Learn/CSS/Introduction_to_CSS/Cascade_and_inheritance

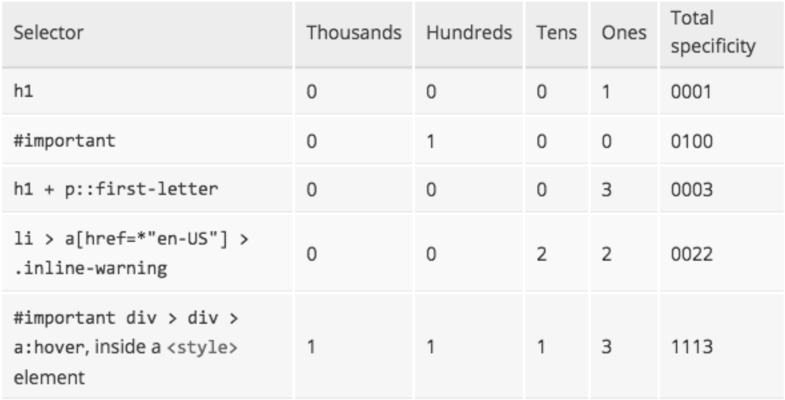
Specificity

The amount of specificity a selector has is measured using four different values (or components), which can be thought of as thousands, hundreds, tens and ones — four single digits in four columns:

- **Thousands:** Score one in this column if the matching selector is inside a <style> element or the declaration is inside a style attribute (such declarations don't have selectors, so their specificity is always simply 1000.) Otherwise 0.
- **Hundreds:** Score one in this column for each ID selector contained inside the overall selector.
- **Tens:** Score one in this column for each class selector, attribute selector, or pseudo-class contained inside the overall selector.
- **Ones:** Score one in this column for each element selector or pseudo-element contained inside the overall selector.

Source: https://developer.mozilla.org/en-US/docs/Learn/CSS/Introduction_to_CSS/Cascade_and_inheritance

Specificity



Selector	Thousands	Hundreds	Tens	Ones	Total specificity
`h1`	0	0	0	1	0001
`#important`	0	1	0	0	0100
`h1 + p::first-letter`	0	0	0	3	0003
`li > a[href=*"en-US"] > .inline-warning`	0	0	2	2	0022
`#important div > div > a:hover`, inside a `<style>` element	1	1	1	3	1113

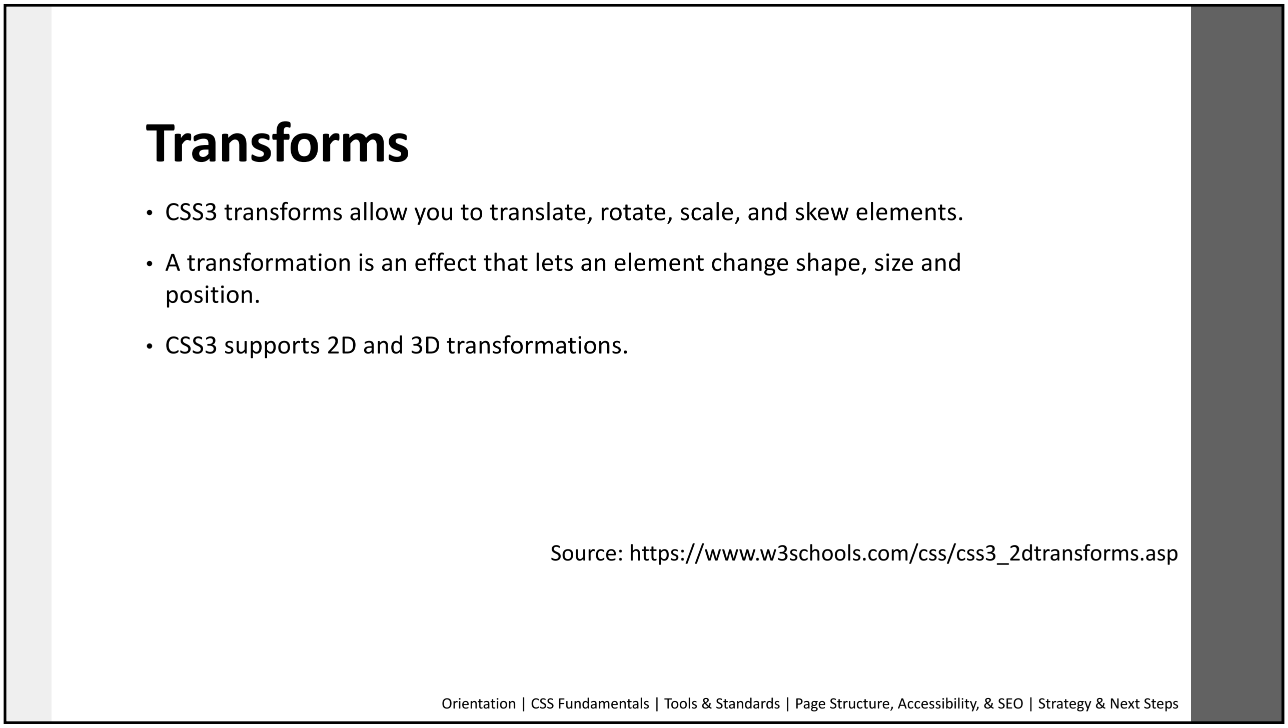
Source: https://developer.mozilla.org/en-US/docs/Learn/CSS/Introduction_to_CSS/Cascade_and_inheritance

Selectors

https://en.wikipedia.org/wiki/Cascading_Style_Sheets#Selector

Transforms // Transitions // Animations

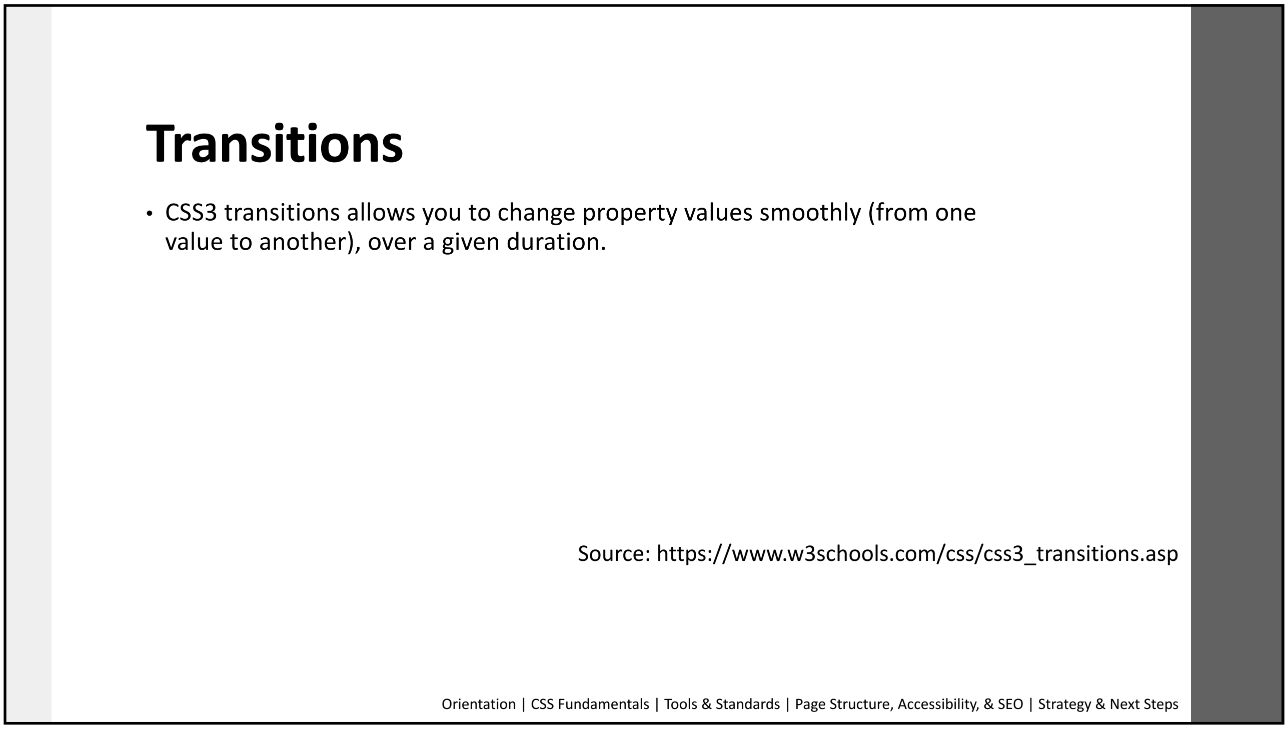
Source: https://www.w3schools.com/css/css3_2dtransforms.asp



Transforms

- CSS3 transforms allow you to translate, rotate, scale, and skew elements.
- A transformation is an effect that lets an element change shape, size and position.
- CSS3 supports 2D and 3D transformations.

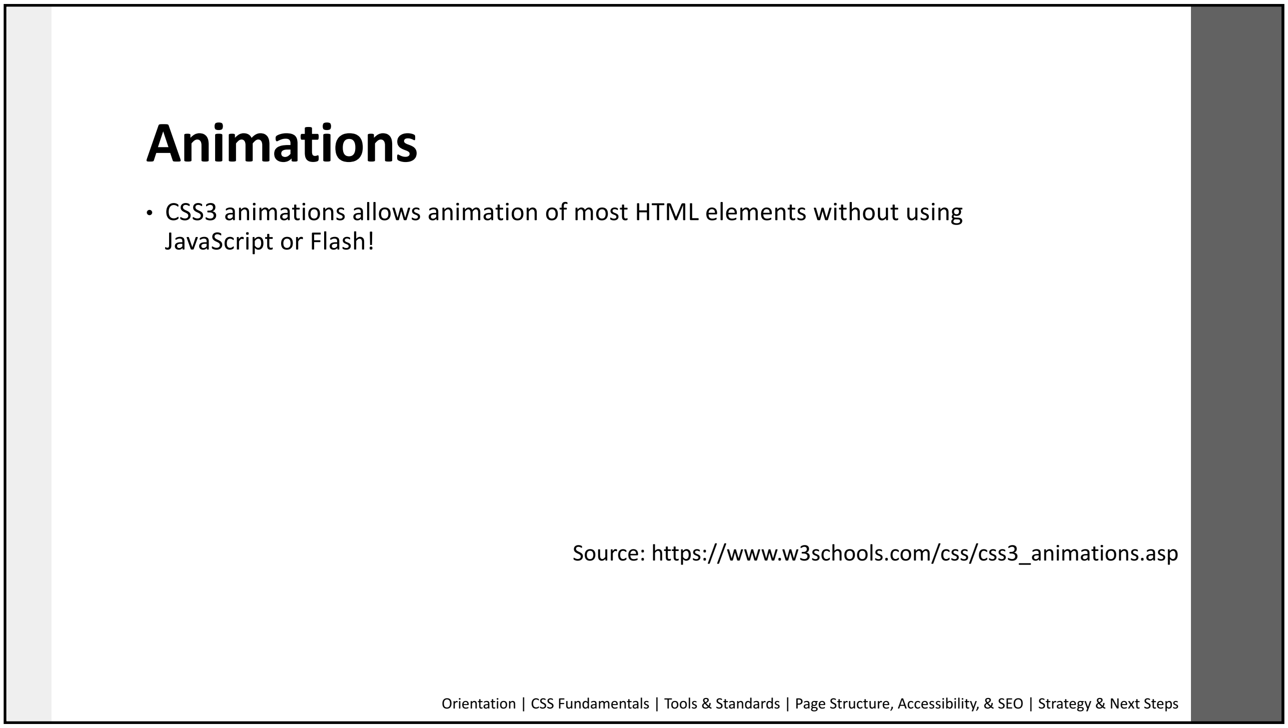
Source: https://www.w3schools.com/css/css3_2dtransforms.asp



Transitions

- CSS3 transitions allows you to change property values smoothly (from one value to another), over a given duration.

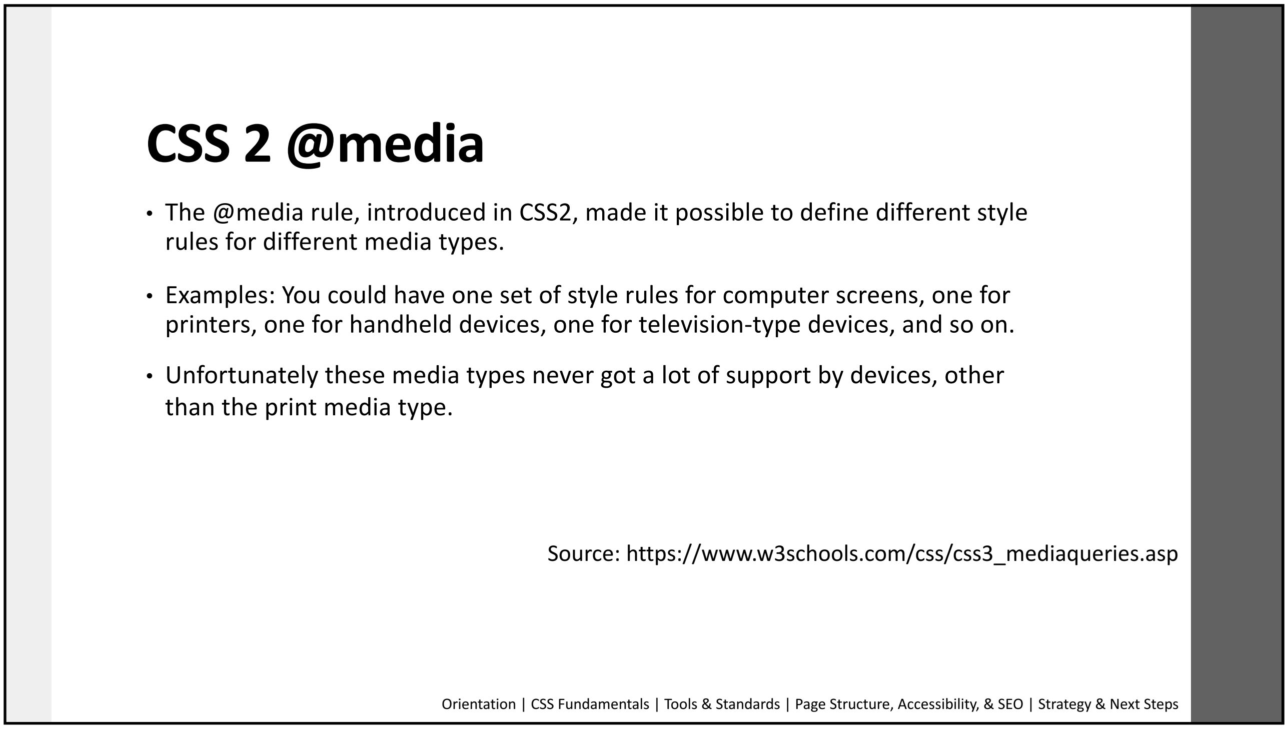
Source: https://www.w3schools.com/css/css3_transitions.asp



Animations

- CSS3 animations allows animation of most HTML elements without using JavaScript or Flash!

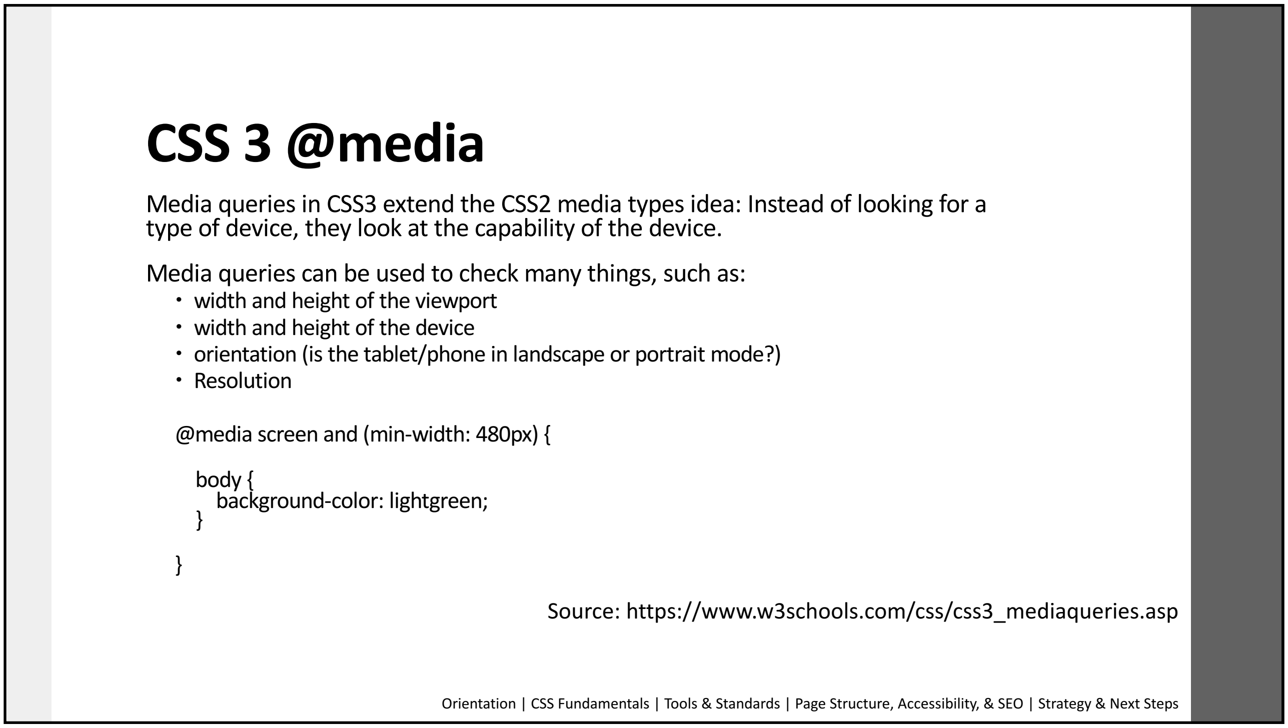
Source: https://www.w3schools.com/css/css3_animations.asp



CSS 2 @media

- The @media rule, introduced in CSS2, made it possible to define different style rules for different media types.
- Examples: You could have one set of style rules for computer screens, one for printers, one for handheld devices, one for television-type devices, and so on.
- Unfortunately these media types never got a lot of support by devices, other than the print media type.

Source: https://www.w3schools.com/css/css3_mediaqueries.asp



CSS 3 @media

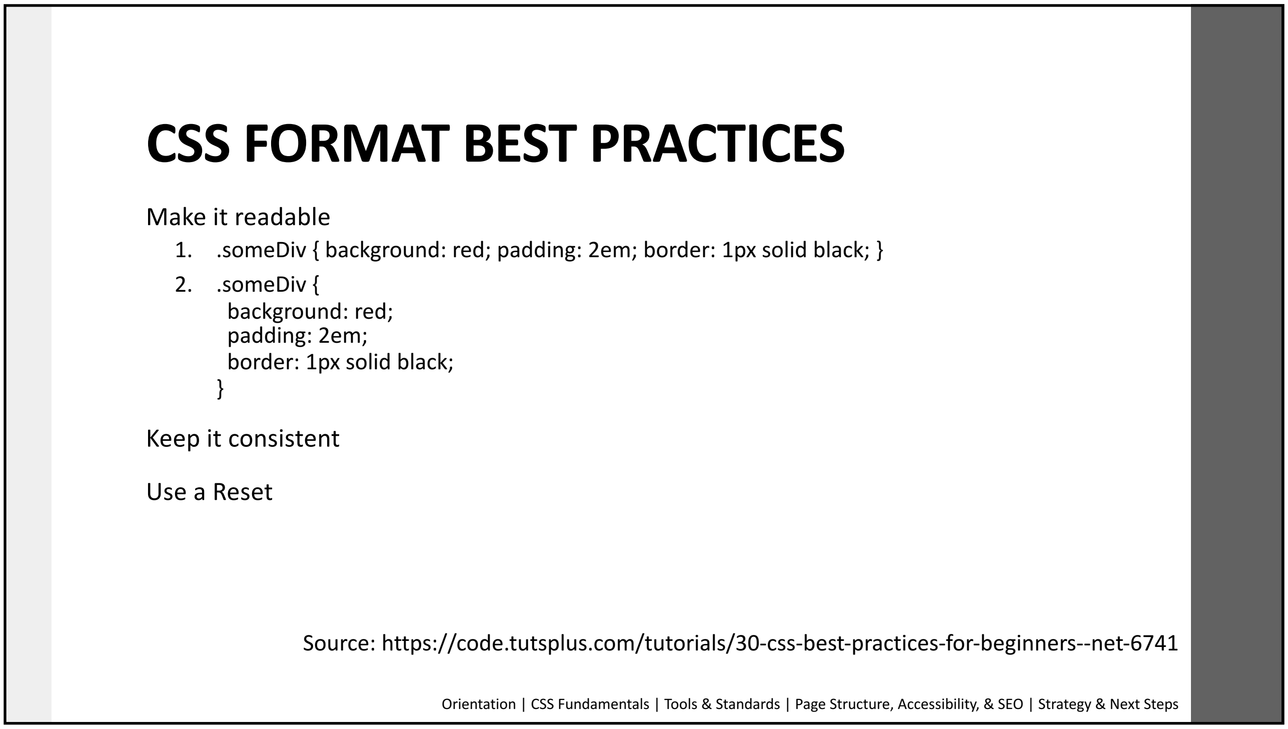
Media queries in CSS3 extend the CSS2 media types idea: Instead of looking for a type of device, they look at the capability of the device.

Media queries can be used to check many things, such as:

- width and height of the viewport
- width and height of the device
- orientation (is the tablet/phone in landscape or portrait mode?)
- Resolution

```
@media screen and (min-width: 480px) {

    body {
        background-color: lightgreen;
    }

}
```

Source: https://www.w3schools.com/css/css3_mediaqueries.asp



CSS FORMAT BEST PRACTICES

Make it readable

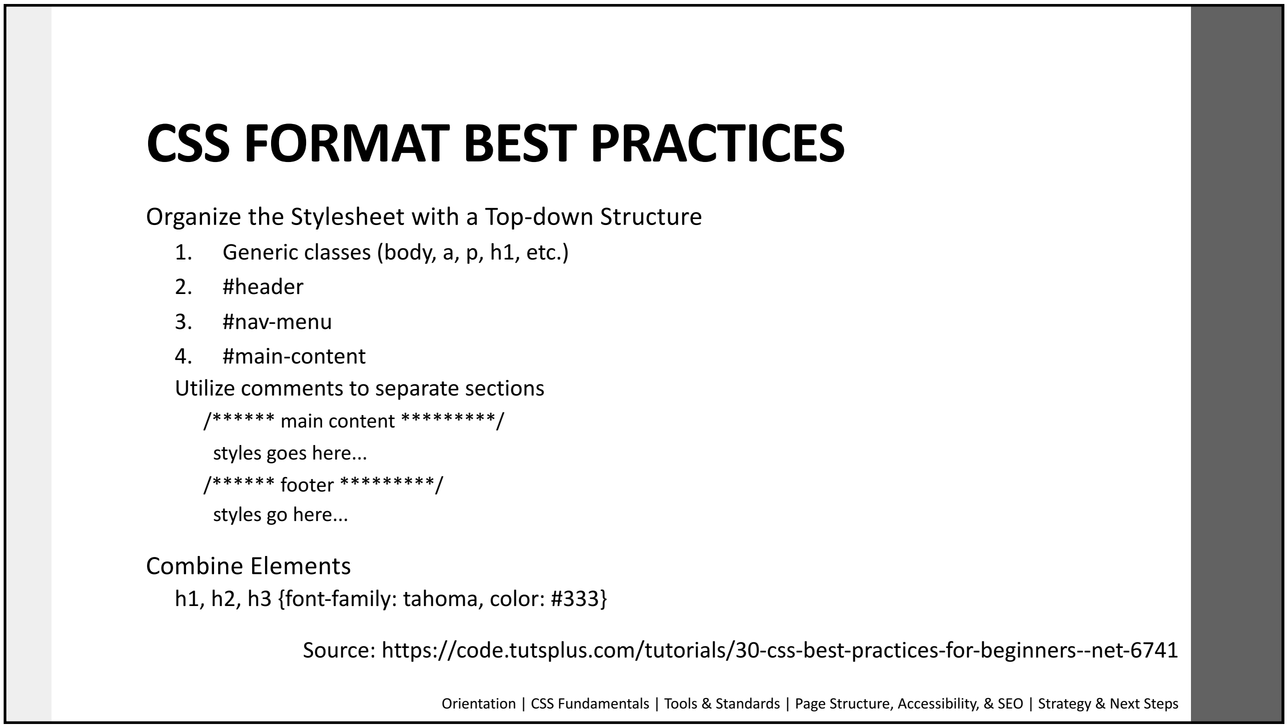
1. .someDiv { background: red; padding: 2em; border: 1px solid black; }
2. .someDiv {

```
  background: red;
  padding: 2em;
  border: 1px solid black;
}
```

Keep it consistent

Use a Reset

Source: https://code.tutsplus.com/tutorials/30-css-best-practices-for-beginners--net-6741



CSS FORMAT BEST PRACTICES

Organize the Stylesheet with a Top-down Structure

1. Generic classes (body, a, p, h1, etc.)
2. #header
3. #nav-menu
4. #main-content

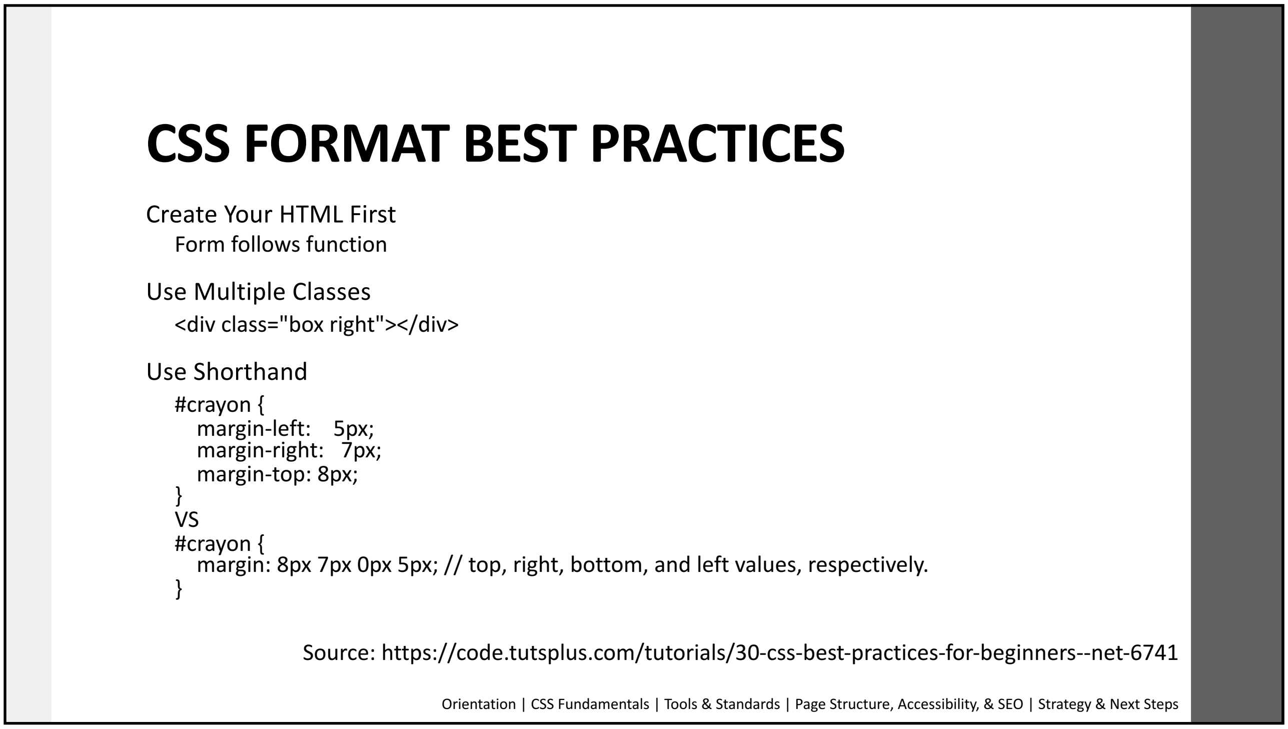
Utilize comments to separate sections

```
/****** main content *********/
  styles goes here...
/****** footer *********/
  styles go here...
```

Combine Elements

```
h1, h2, h3 {font-family: tahoma, color: #333}
```

Source: https://code.tutsplus.com/tutorials/30-css-best-practices-for-beginners--net-6741



CSS FORMAT BEST PRACTICES

Create Your HTML First

Form follows function

Use Multiple Classes

```
<div class="box right"></div>
```

Use Shorthand

```
#crayon {
   margin-left:    5px;
   margin-right:   7px;
   margin-top: 8px;
}
VS
#crayon {
   margin: 8px 7px 0px 5px; // top, right, bottom, and left values, respectively.
}
```

Source: https://code.tutsplus.com/tutorials/30-css-best-practices-for-beginners--net-6741

CSS FORMAT BEST PRACTICES

Comment your CSS

Make Use of Generic Classes

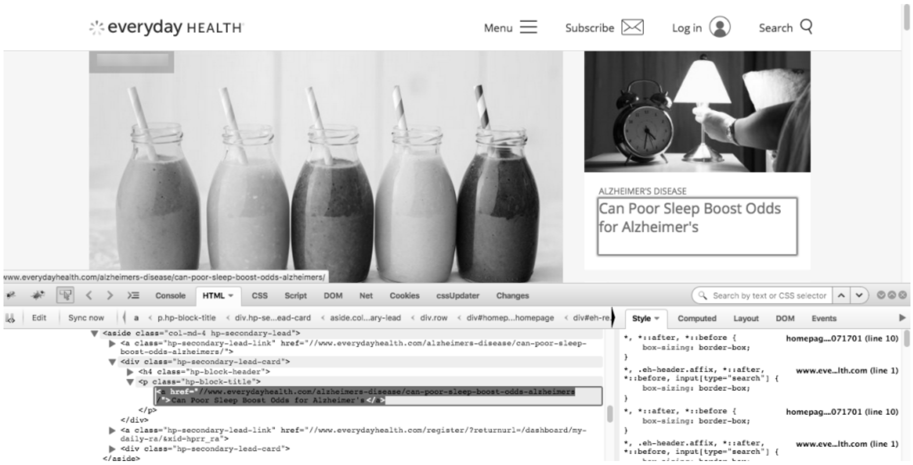
Avoid Extra Selectors

```
body #container .someclass ul li {....}
VS
.someclass li {...}
```

VALIDATE

Source: https://code.tutsplus.com/tutorials/30-css-best-practices-for-beginners--net-6741

The Inspector

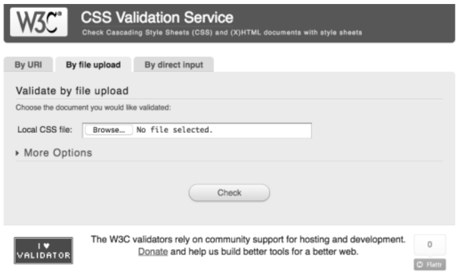


```
▼ <aside class="col-md-4 hp-secondary-lead">
  ▶ <a class="hp-secondary-lead-link" href="//www.everydayhealth.com/alzheimers-disease/can-poor-sleep-
    boost-odds-alzheimers/">
  ▼ <div class="hp-secondary-lead-card">
    ▶ <h4 class="hp-block-header">
    ▼ <p class="hp-block-title">
        <a href="//www.everydayhealth.com/alzheimers-disease/can-poor-sleep-boost-odds-alzheimers
        /">Can Poor Sleep Boost Odds for Alzheimer's</a>
      </p>
    </div>
  ▶ <a class="hp-secondary-lead-link" href="//www.everydayhealth.com/register/?returnurl=/dashboard/my-
    daily-ra/&xid-hprr_ra">
  ▶ <div class="hp-secondary-lead-card">
  </aside>
```

```
*, *::after, *::before {                     homepag....071701 (line 10)
    box-sizing: border-box;
}
*, .eh-header.affix, *::after,                 www.eve...lth.com (line 1)
*::before, input[type="search"] {
    box-sizing: border-box;
}
*, *::after, *::before {                     homepag....071701 (line 10)
    box-sizing: border-box;
}
*, .eh-header.affix, *::after,                 www.eve...lth.com (line 1)
*::before, input[type="search"] {
```

W3C Validation Tool

https://jigsaw.w3.org/css-validator/



CSS Validation Service

Check Cascading Style Sheets (CSS) and (X)HTML documents with style sheets

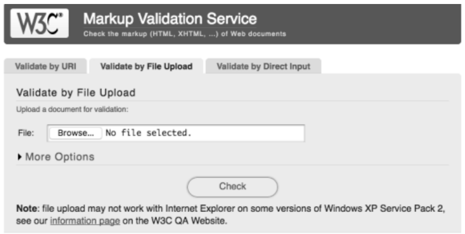
Validate by file upload

Choose the document you would like validated:

Local CSS file: No file selected.

The W3C validators rely on community support for hosting and development. Donate and help us build better tools for a better web.

https://validator.w3.org



Markup Validation Service

Check the markup (HTML, XHTML, ...) of Web documents

Validate by File Upload

Upload a document for validation:

File: No file selected.

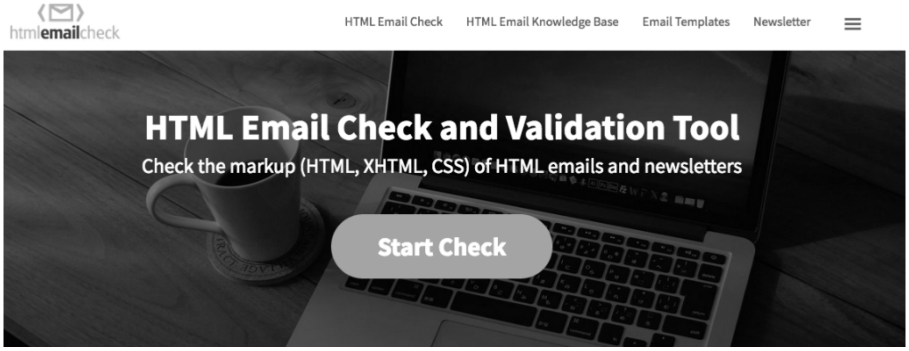
Note: file upload may not work with Internet Explorer on some versions of Windows XP Service Pack 2, see our information page on the W3C QA Website.

Be nice: validate

- Validation as a debugging tool
- Validation as a future-proof quality check
- Validation eases maintenance
- Validation helps teach good practices
- Validation is a sign of professionalism

Source: https://validator.w3.org/docs/why.html

HTML Email Check



HTML Email Check and Validation Tool

Check the markup (HTML, XHTML, CSS) of HTML emails and newsletters

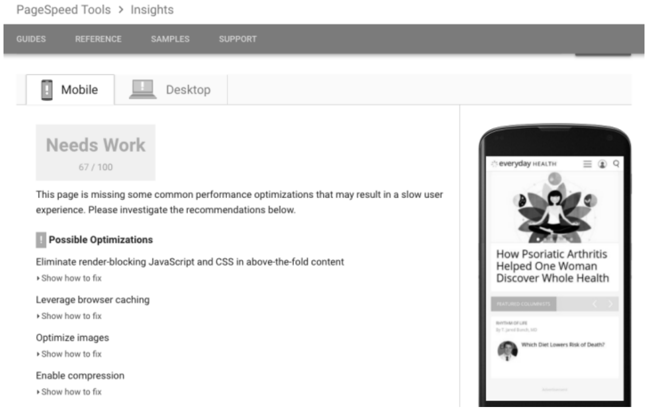
Source: https://www.htmlemailcheck.com/

Speed is important

If an e-commerce site is making $100,000 per day,
a 1 second page delay could potentially cost
$2.5 million in lost sales every year.

Source: https://blog.kissmetrics.com/loading-time/

PageSpeed Insights


PageSpeed Tools > Insights
GUIDES
REFERENCE
SAMPLES
SUPPORT
Mobile
Desktop
everyday HEALTH
How Psoriatic Arthritis Helped One Woman Discover Whole Health
FEATURED COLUMNISTS
Which Diet Lowers Risk of Death?

Needs Work

67 / 100

This page is missing some common performance optimizations that may result in a slow user experience. Please investigate the recommendations below.

! Possible Optimizations

Eliminate render-blocking JavaScript and CSS in above-the-fold content
- Show how to fix

Leverage browser caching
- Show how to fix

Optimize images
- Show how to fix

Enable compression
- Show how to fix

https://developers.google.com/speed/pagespeed/insights/

Hierarchy

Shows

Where to look

and when

Machines utilize hierarchy too

Search Engine Optimization

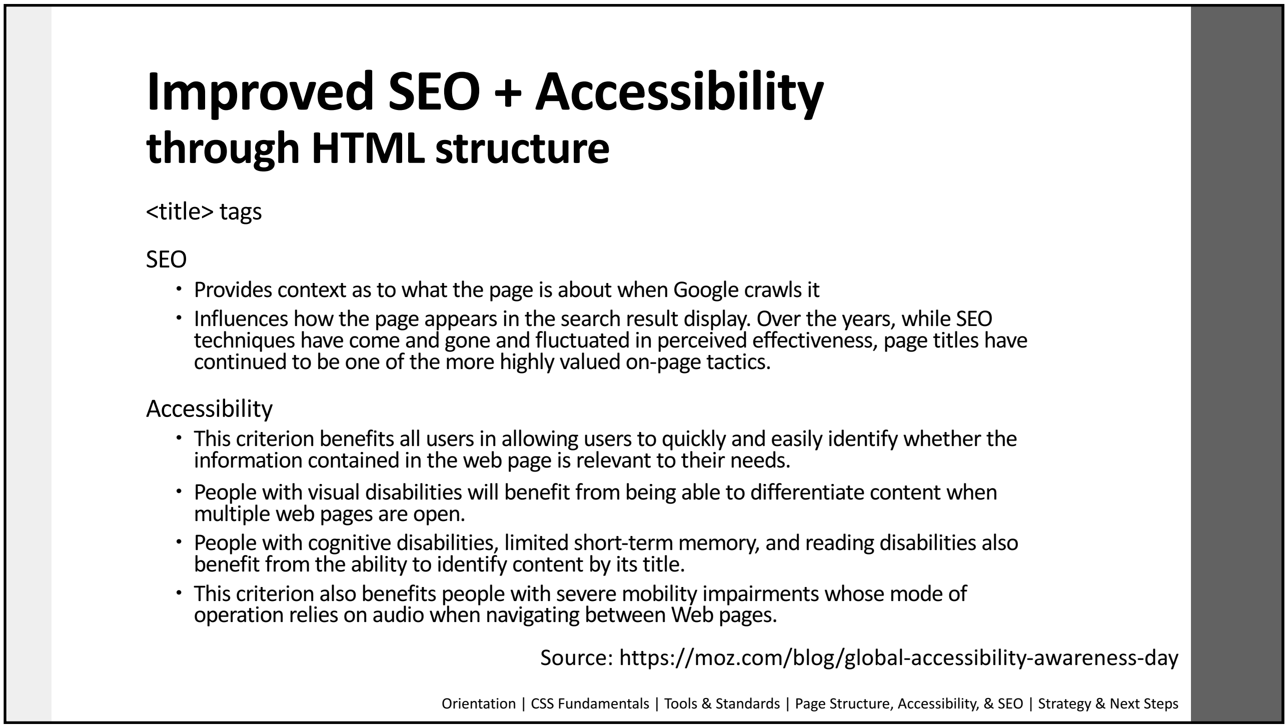
SEO is a marketing discipline focused on growing visibility in organic (non-paid) search engine results. SEO encompasses both the technical and creative elements required to improve rankings, drive traffic, and increase awareness in search engines. There are many aspects to SEO, from the words on your page to the way other sites link to you on the web. Sometimes SEO is simply a matter of making sure your site is structured in a way that search engines understand.

Source: https://moz.com/beginners-guide-to-seo

Web Accessibility

Web accessibility means that people with disabilities can use the Web. More specifically, Web accessibility means that people with disabilities can perceive, understand, navigate, and interact with the Web, and that they can contribute to the Web. Web accessibility also benefits others, including older people with changing abilities due to aging.

Source: https://www.w3.org/WAI/intro/accessibility.php



Improved SEO + Accessibility
through HTML structure

<title> tags

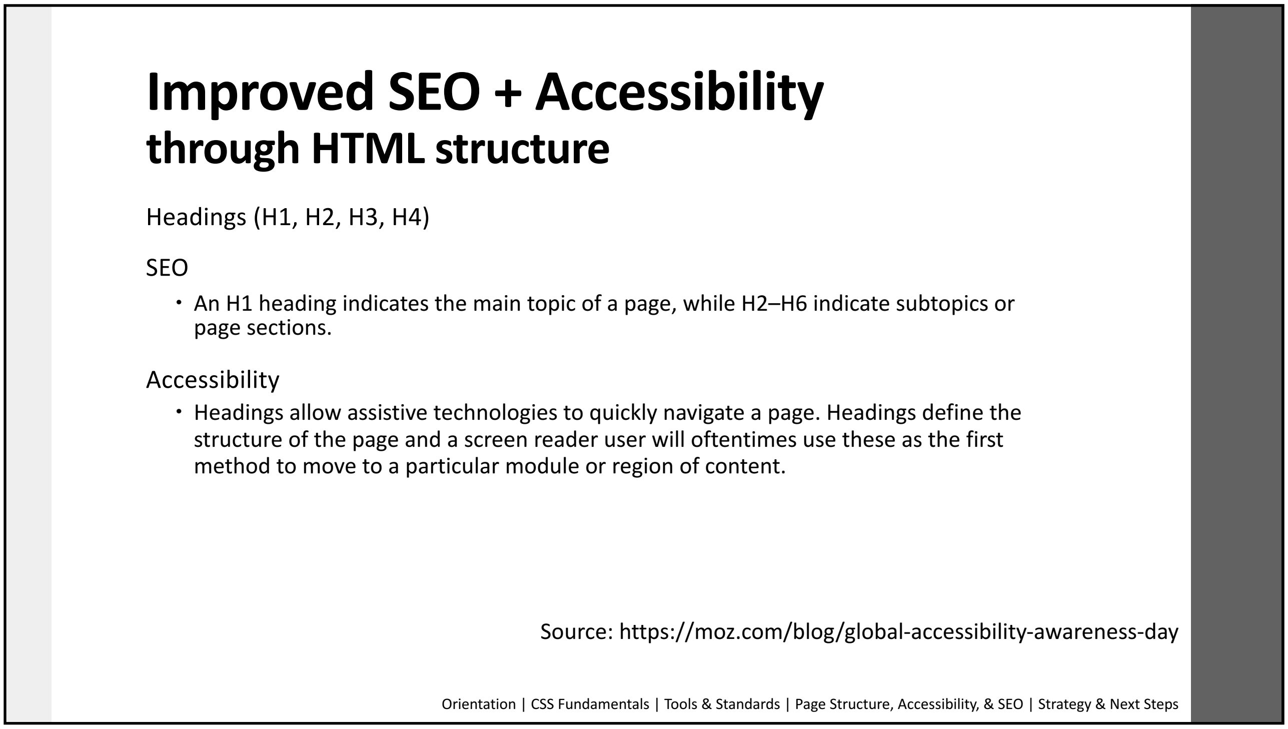
SEO

- Provides context as to what the page is about when Google crawls it
- Influences how the page appears in the search result display. Over the years, while SEO techniques have come and gone and fluctuated in perceived effectiveness, page titles have continued to be one of the more highly valued on-page tactics.

Accessibility

- This criterion benefits all users in allowing users to quickly and easily identify whether the information contained in the web page is relevant to their needs.
- People with visual disabilities will benefit from being able to differentiate content when multiple web pages are open.
- People with cognitive disabilities, limited short-term memory, and reading disabilities also benefit from the ability to identify content by its title.
- This criterion also benefits people with severe mobility impairments whose mode of operation relies on audio when navigating between Web pages.

Source: https://moz.com/blog/global-accessibility-awareness-day



Improved SEO + Accessibility
through HTML structure

Headings (H1, H2, H3, H4)

SEO

- An H1 heading indicates the main topic of a page, while H2–H6 indicate subtopics or page sections.

Accessibility

- Headings allow assistive technologies to quickly navigate a page. Headings define the structure of the page and a screen reader user will oftentimes use these as the first method to move to a particular module or region of content.

Source: https://moz.com/blog/global-accessibility-awareness-day

Improved SEO + Accessibility
through HTML structure

Heading do’s and don’ts

- **Do use headings.** It’s important not to skip headings.
- **Don’t use more than one H1 heading**: HTML5 allows for multiple H1s, but this is not well-supported by browsers/assistive technology.
- **Do use headings to define sections of content.** For example, use H2 headings for subheader or key sections of the page and H3 headings for content modules.
- **Don’t use headings if there's no following content.** Headings define sections of content, so if there is no content section, it shouldn’t have a header.

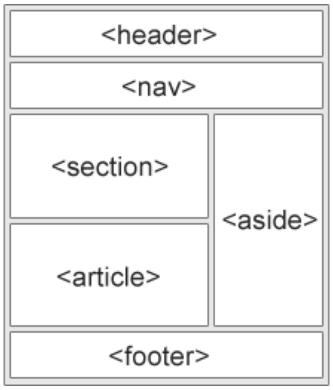
Source: https://moz.com/blog/global-accessibility-awareness-day

Improved SEO + Accessibility
through HTML structure

Utilize the appropriate element tags

- **One H1 per page – should contain information about the page and keywords**
  It is permitted in HTML5 sections to have more than one
- **Headings create divisions in page content**
- **Lists in <ul> or <ol> tags**
- **<strong> and <em> for emphasis**
  not <b> or <i>
- **Add title and alt attributes to all images**
- **Content will be read by spiders and screen readers in the order it’s coded**
  Place the most important information toward the top
- **Organize pages with HTML5 semantic elements**
  <header>, <main>, <section>, <aside>, <footer>, etc.

Improved SEO + Accessibility
through HTML structure



<header>

<nav>

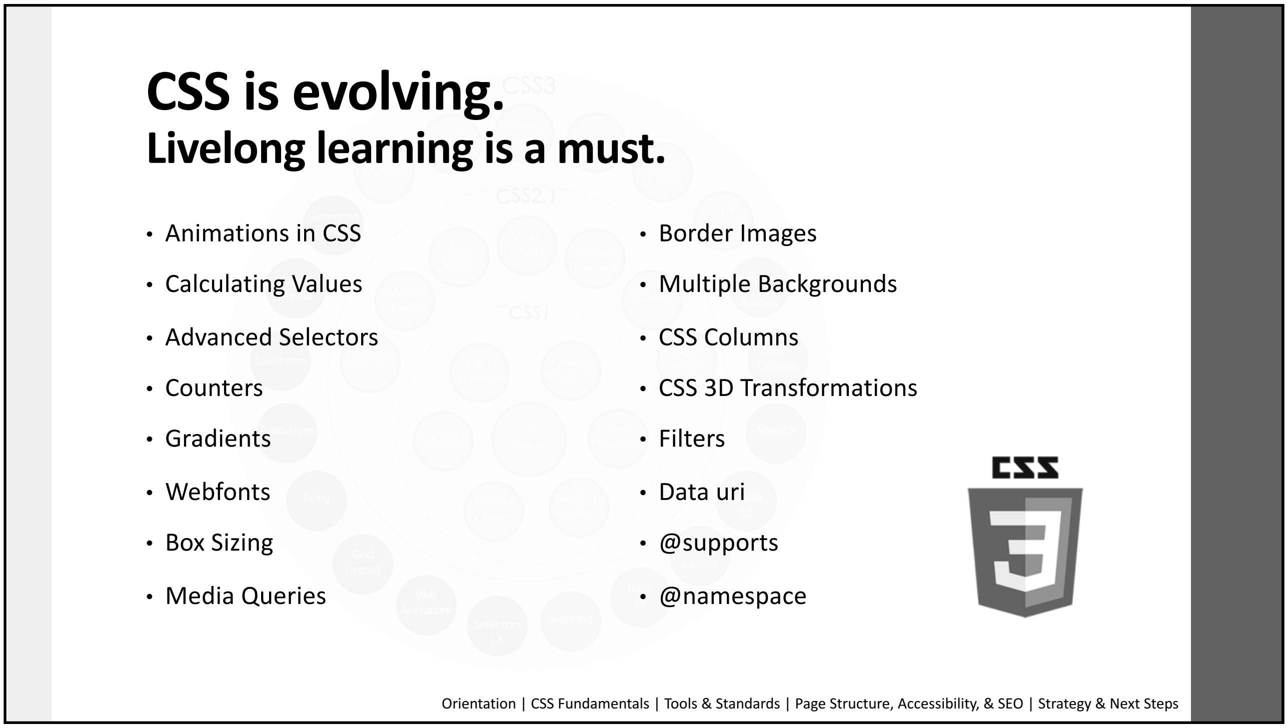
<section>

<article>

<aside>

<footer>

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



CSS is evolving.
Livelong learning is a must.

- Animations in CSS
- Calculating Values
- Advanced Selectors
- Counters
- Gradients
- Webfonts
- Box Sizing
- Media Queries
- Border Images
- Multiple Backgrounds
- CSS Columns
- CSS 3D Transformations
- Filters
- Data uri
- @supports
- @namespace

Orientation | CSS Fundamentals | Tools & Standards | Page Structure, Accessibility, & SEO | Strategy & Next Steps

CSS is evolving.
Livelong learning is a must.

https://davidwalsh.name/css4-preview
https://cloudfour.com/thinks/building-themes-with-css4-color-features/
https://www.w3.org/TR/2011/WD-selectors4-20110929/
https://www.w3.org/

Orientation | CSS Fundamentals | Tools & Standards | Page Structure, Accessibility, & SEO | Strategy & Next Steps