LET'S GET EVERYTHING SET UP!

1. In Slack, join the 02-CSS-Intro channel.
2. Download the Lesson 2 starter code and slides.
3. Move the starter code and slides from your Downloads folder to the **fewd** folder we added to your Documents folder last class.
4. Double-click on starter code folder to unzip it
5. After you've unzipped, be sure to delete the original .zip file to avoid confusion and make sure you don't unzip it again later!
6. Open the entire **fewd** folder with your code editor (such as Visual Studio Code) (either drag and drop the folder on the Code icon in the dock on Mac, or open Visual Studio Code, click File > Open, and select the starter code folder.

WEEKLY OVERVIEW

WEEK 1	HTML Basics / CSS Basics
WEEK 2	CSS Box Model / CSS Layout
WEEK 3	Layout Lab / Comp to Web Lab

LEARNING OBJECTIVES

- Be able to describe relationships between elements (parent, child, ancestor, descendent, etc.)
- Add an external stylesheet to an HTML web page.
- Predict image paths and apply relative paths to <img> and <a> tags.
- Use CSS to add basic styles to a website
- Recognize, interpret, and apply various CSS color formats
- Manipulate the border, padding, and margin around content using CSS.

AGENDA

- Review
- More HTML Basics
- Intro to CSS
- Lab Time

Checkin and questions

- The **most significant thing I learned** about basic HTML is _____.
- My **biggest outstanding question** about basic HTML is _____.

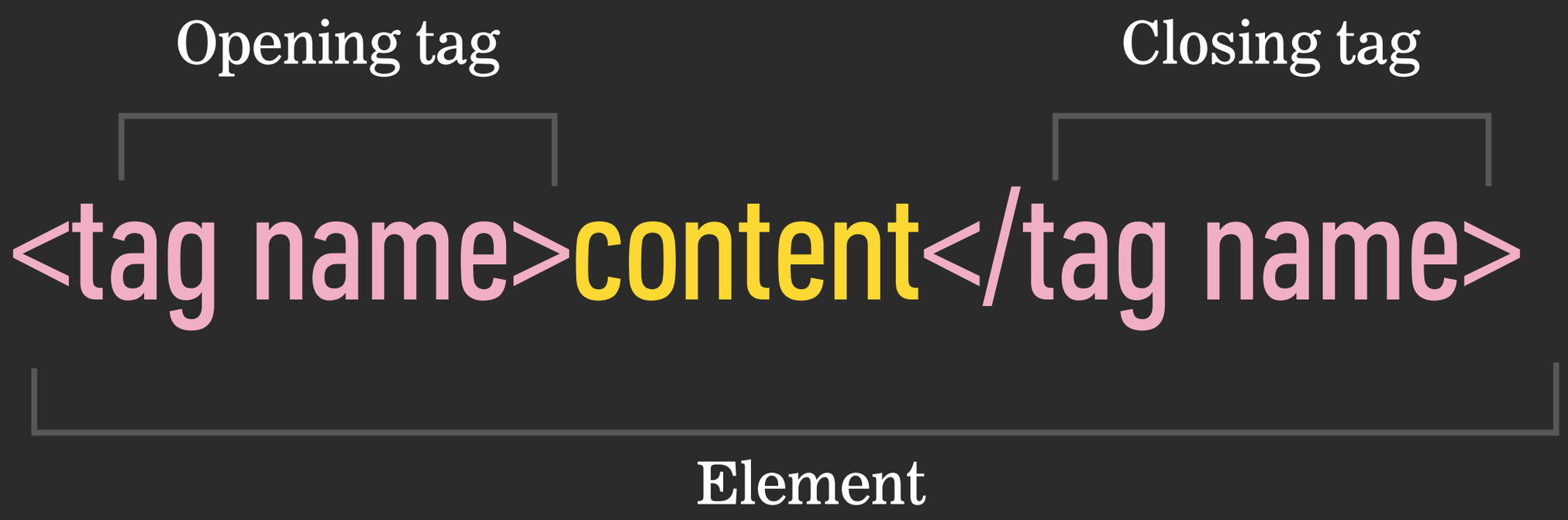
LET'S GET THINKING

HOW CAN YOU CHANGE THE STYLE OF ITEMS ON A WEB PAGE?

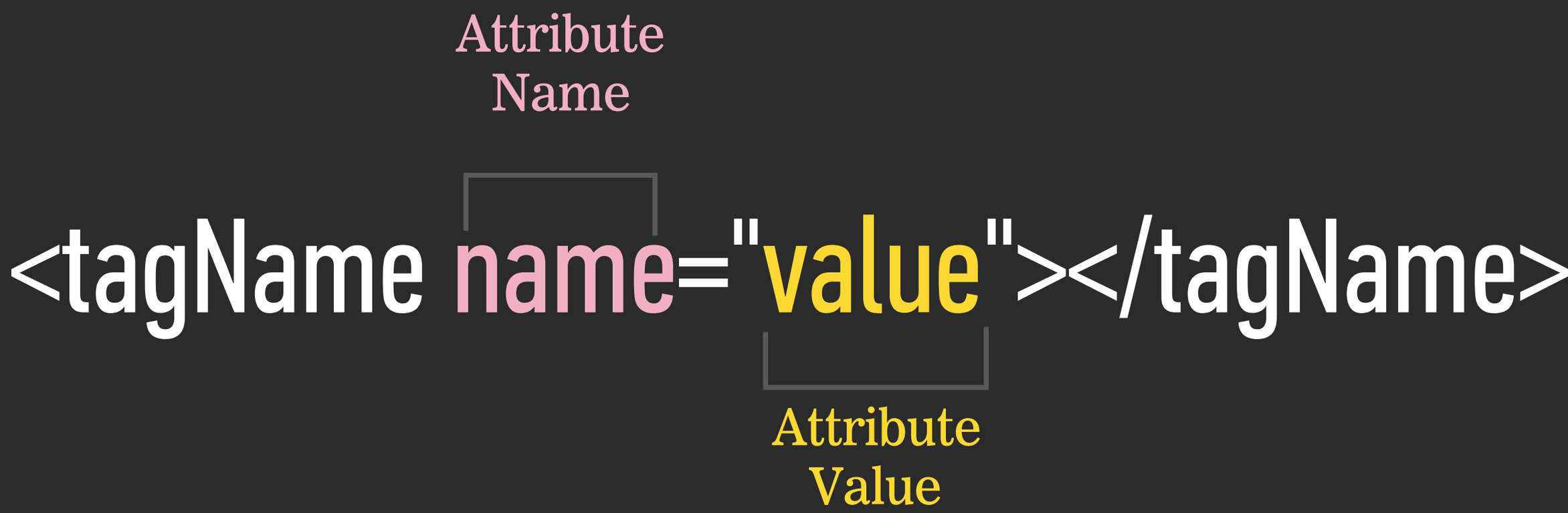
- Which ways have we looked at already?
- Which other ways have you seen?

CSS BASICS

HTML SYNTAX — TAGS



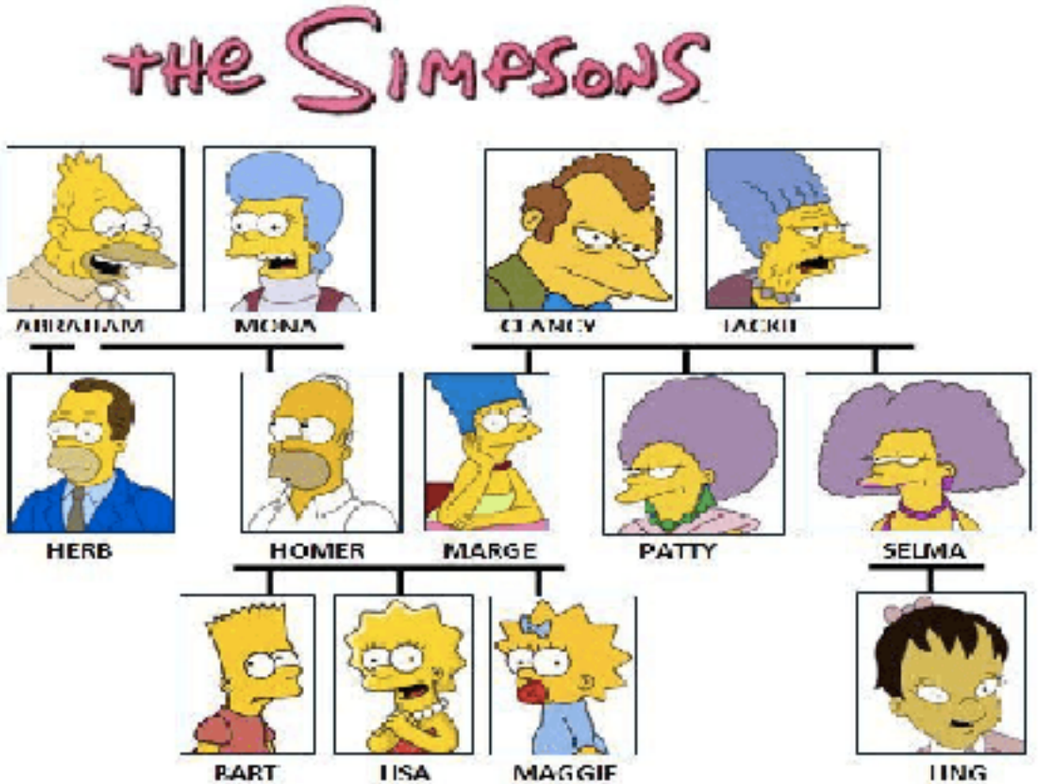
HTML SYNTAX — ATTRIBUTES



HTML STRUCTURE

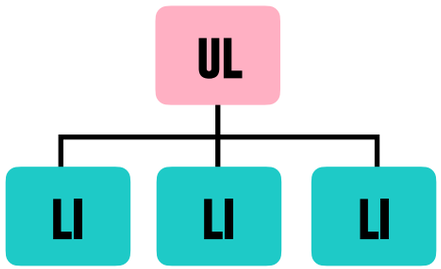
PAGE STRUCTURE — RELATIONSHIPS

- We often use the same terms we would use to describe relationships between family members to describe the relationships between HTML elements.



PAGE STRUCTURE — NESTING

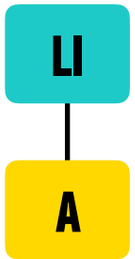
- Here we can say that our <ul> is the **parent** of our <li>s
- We can also say that our <li>s are **children** of the <ul>




```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>The Evolution of Denim</title>
</head>
<body>
   <h1>The Evolution of Denim</h1>
    <ul>
        <li>Dark Wash <a href="linkgoeshere">Jeans</a></li>
        <li>Stone Wash</li>
        <li>Chambray</li>
    </ul>
</body>
</html>
```

PAGE STRUCTURE — NESTING

- Similarly, we can have an <a> tag that is nested inside, or wrapped by, our <li> element.



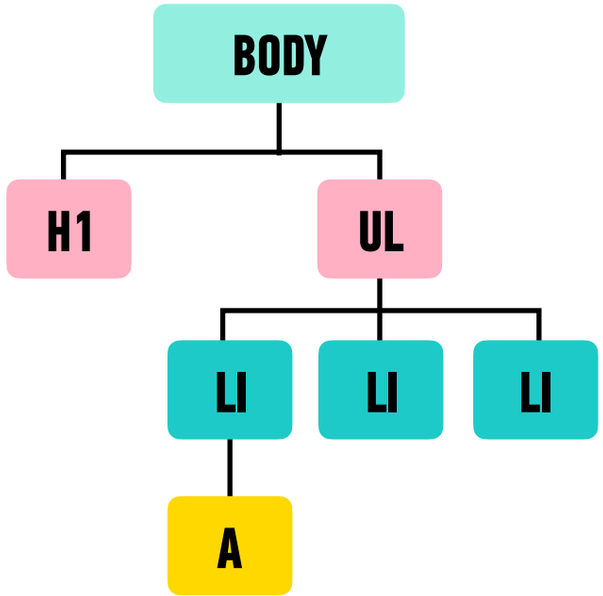

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>The Evolution of Denim</title>
</head>
<body>
    <h1>The Evolution of Denim</h1>
    <ul>
        <li>Dark Wash <a href="linkgoeshere">Jeans</a></li>
        <li>Stone Wash</li>
        <li>Chambray</li>
    </ul>
</body>
</html>
```

<a>

<li>

PAGE STRUCTURE — NESTING

- Similarly, we can say that all of our HTML content, our h1, our ul, our li elements, are "wrapped" by the body, or "nested" inside the body since they are within the opening and closing body tags

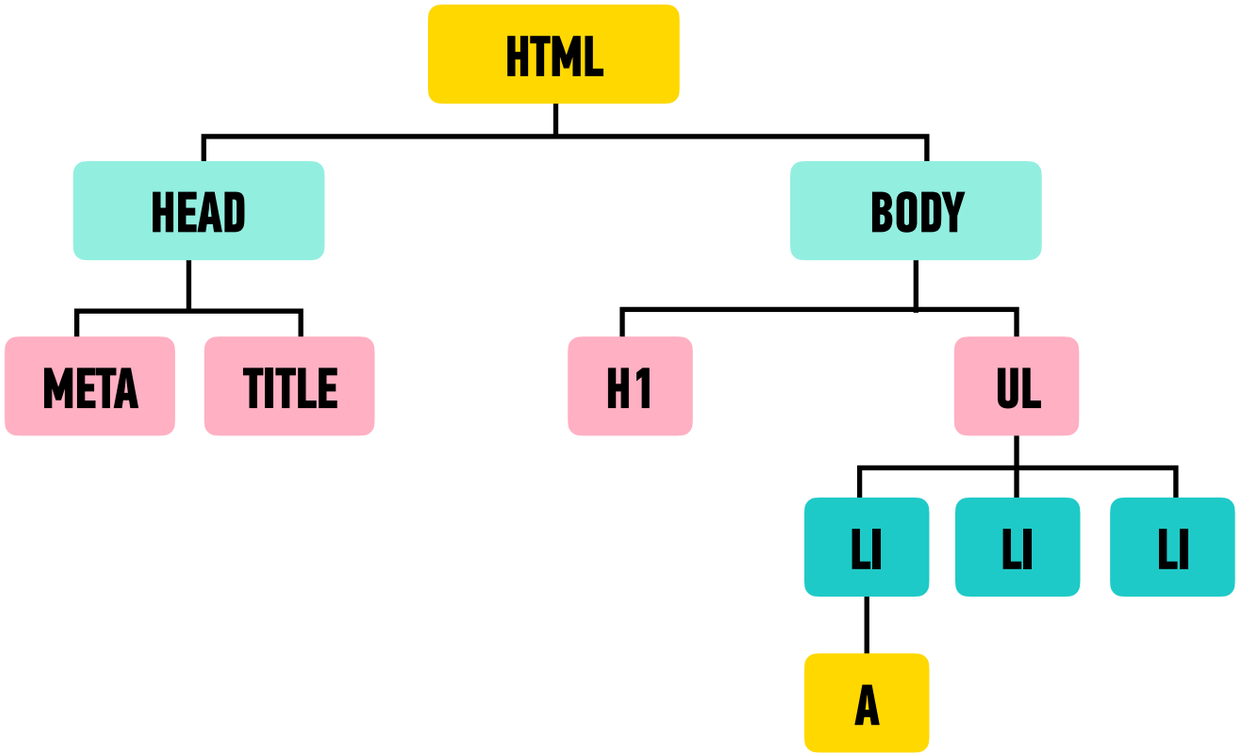


body

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>The Evolution of Denim</title>
</head>
<body>
    <h1>The Evolution of Denim</h1>
    <ul>
        <li>Dark Wash <a href="linkgoeshere">Jeans</a></li>
        <li>Stone Wash</li>
        <li>Chambray</li>
    </ul>
</body>
</html>
```

HTML

DOM TREE




```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>The Evolution of Denim</title>
</head>
<body>
    <h1>The Evolution of Denim</h1>
    <ul>
        <li>Dark Wash <a href="linkgoeshere">Jeans</a></li>
        <li>Stone Wash</li>
        <li>Chambray</li>
    </ul>
</body>
</html>
```

ACTIVITY

EXERCISE

KEY OBJECTIVE

- Be able to describe relationships between elements.

LOCATION

- Starter code > dom_tree

TIMING

2 min

1. Which elements are "parents" of other elements? What are their "children"? Descendants?
2. Which Elements are "children"? What are their parents? What are their ancestors?

PAGE STRUCTURE — INDENT!

- It is best practice to indent any tags that are nested inside other tags.
- This makes our code easier to read and will help us see if we forget a closing tag.
- Editors include features to organize your indents for you

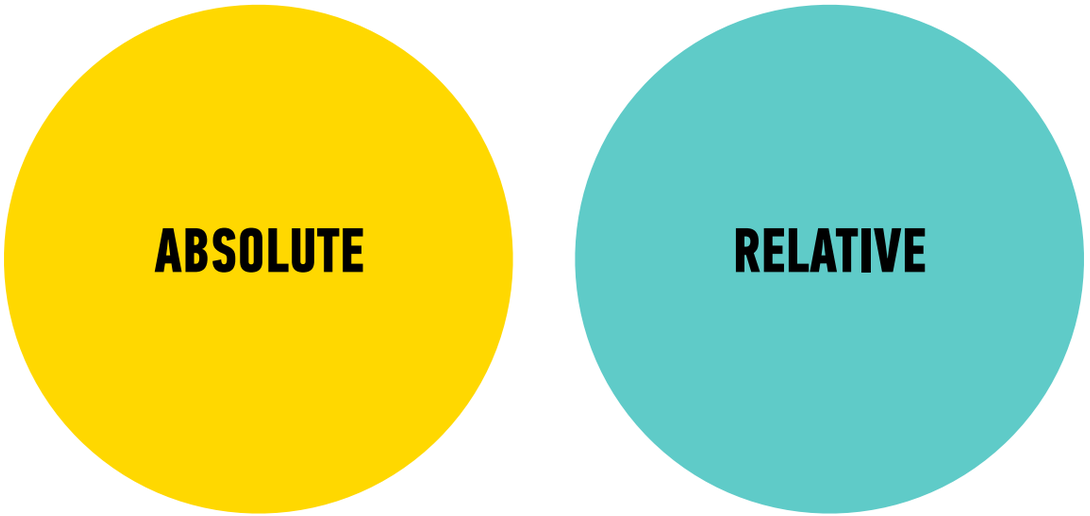
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>The Evolution of Denim</title>
</head>
<body>
    <h1>The Evolution of Denim</h1>
    <ul>
        <li>Dark Wash <a href="linkgoeshere">Jeans</a></li>
        <li>Stone Wash</li>
        <li>Chambray</li>
    </ul>
</body>
</html>
```

CODE ALONG

URLS

TYPES OF URLS

- There are two main types of URLs:



LINKING TO OTHER SITES – ABSOLUTE URLS

ABSOLUTE

WHEN YOU LINK TO ANOTHER SITE:

- Value of the href attribute will be the *full web address* for the site
- This is known as the **absolute** URL.

Absolute URL

```
<a href="http://www.amazon.com">Amazon</a>
```

```
<img src="http://i.imgur.com/vKA0ld0.gif" alt="Guess who?">
```

LINKING TO OTHER PAGES ON THE SAME SITE — RELATIVE URLS

RELATIVE

WHEN YOU LINK TO ANOTHER PAGE OR FILE ON THE SAME SITE:

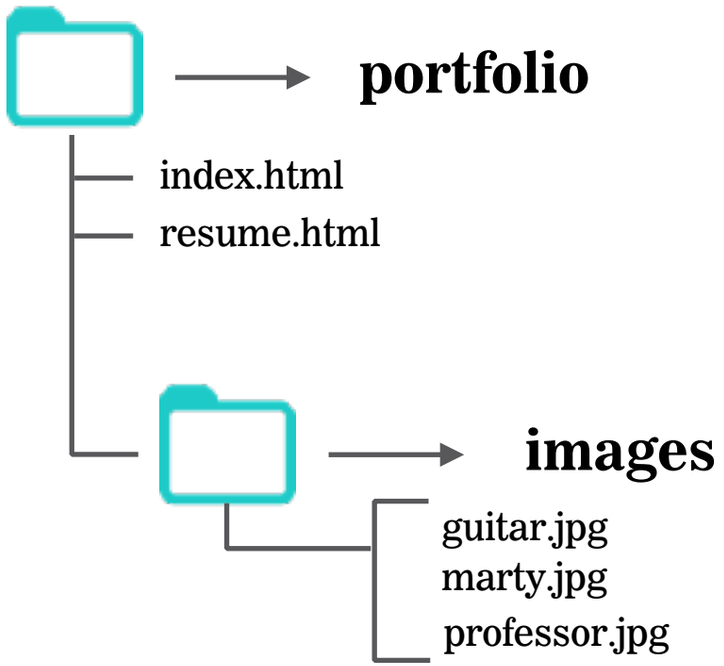
- You don't need to specify a domain name in the URL.
- Use **Relative URL:** indicates where pages are *in relation to the current page*

Relative URL

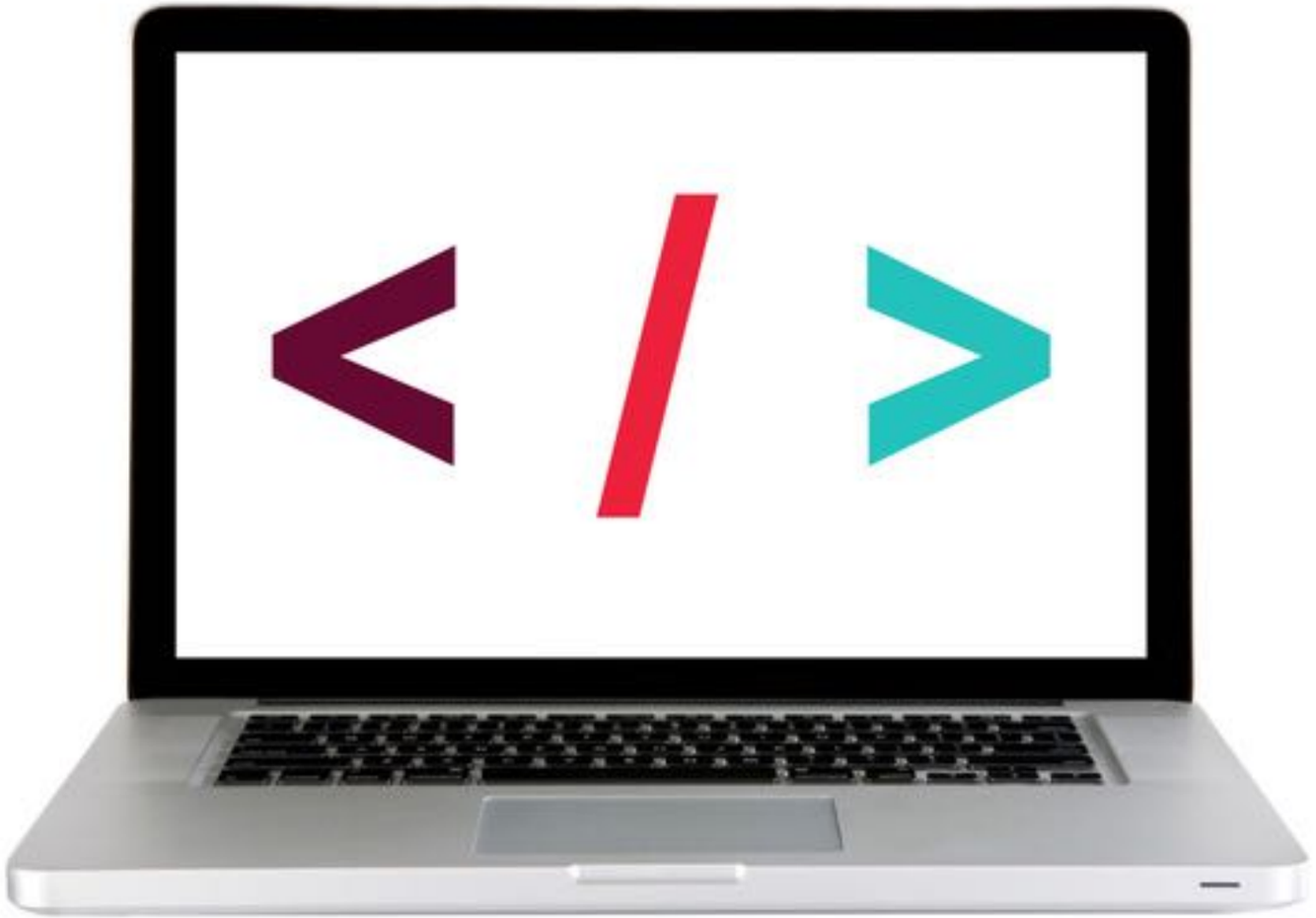
```
<a href="about.html">About Page</a>

<img src="images/logo.png" alt="logo">
```

DIRECTORY STRUCTURE

- On larger sites, it's *best practice* to organize code by placing the files for each major section of the site into a new folder
- These folders are often referred to as **directories**.



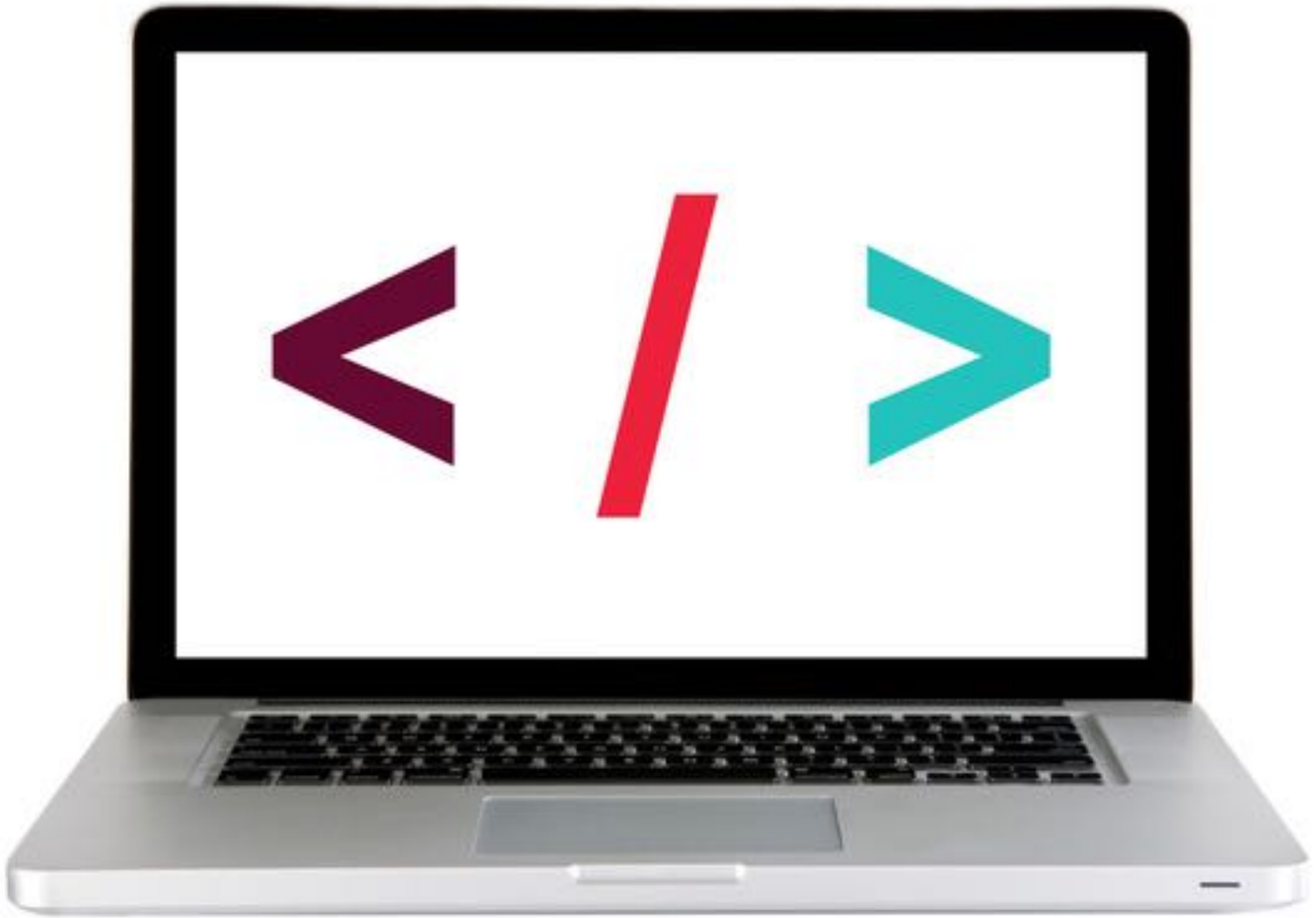
LET'S TAKE A CLOSER LOOK



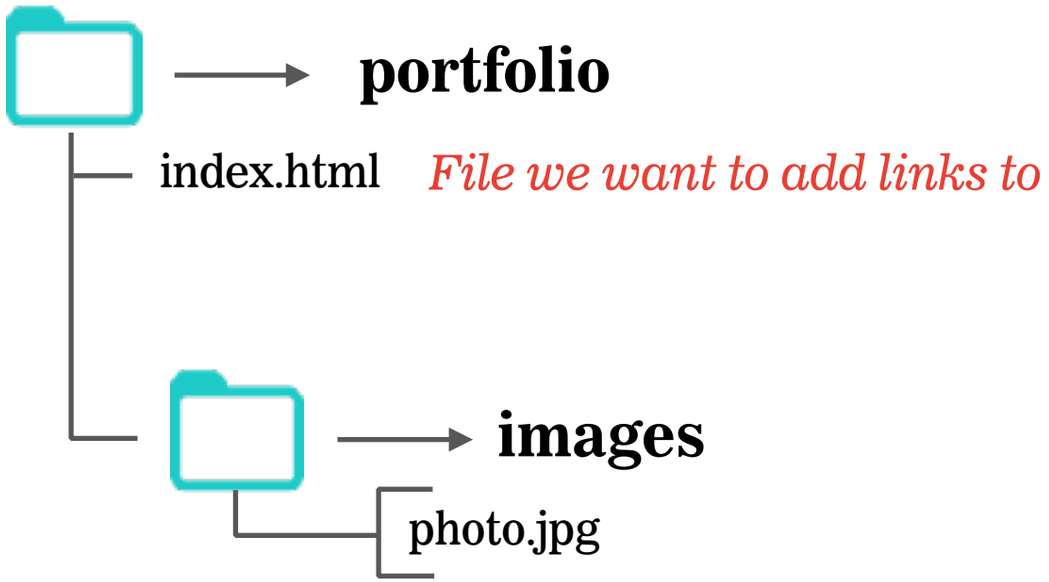
RELATIVE URLS — SAME FOLDER

	RELATIVE URL	EXAMPLE
SAME FOLDER	File name	<a href="info.html">Info</a>
		<img src="photo.jpg" alt="">

LET'S TAKE A CLOSER LOOK

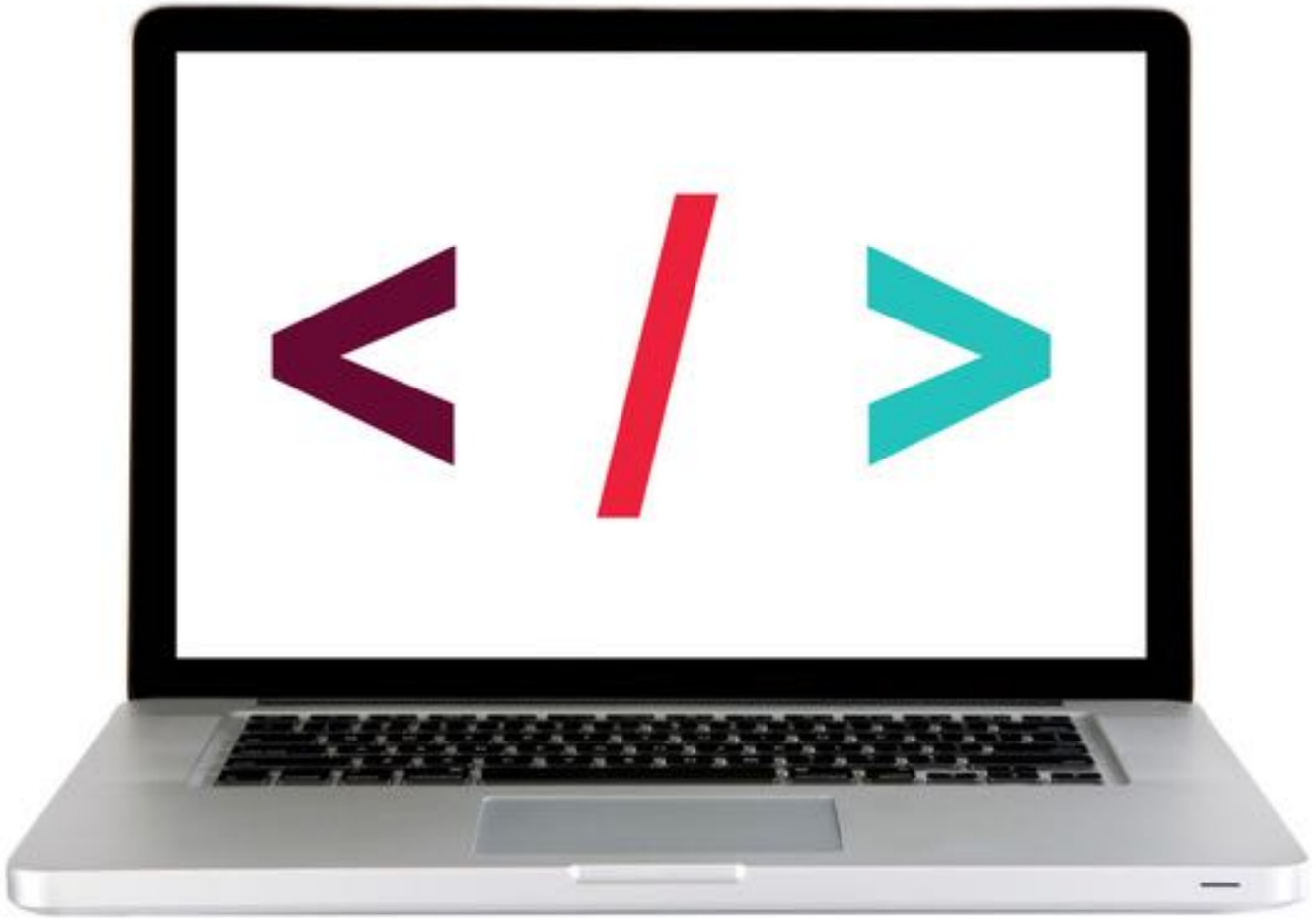


RELATIVE URLS — CHILD FOLDER



	DESCRIPTION	EXAMPLE
CHILD	Name of child folder + / + file name	<img src="images/photo.jpg">
		<a href="about/about.html">About</a>

LET'S TAKE A CLOSER LOOK



ACTIVITY

EXERCISE

KEY OBJECTIVE

- Practice adding relative URLs to a project

KEY OBJECTIVE

- Starter code > Portfolio Folder

TIMING

4 min

1. With a partner — Follow steps 1 - 3 in resume.html
2. Test in browser!

RELATIVE URLS

FOLDER FILE IS IN:	DESCRIPTION	EXAMPLE
SAME	File name	info.html
CHILD	Name of child folder + / + file name	people/index.html
GRANDCHILD	Name of child folder + / + Name of grandchild folder + / + file name	people/culture/index.html
PARENT	../ + path	../index.html
GRANDPARENT	../../ + path	../../index.html

Note that ../ means to go up one directory, and can be used repeatedly:
../../ would go up two directories.

INTRO TO CSS

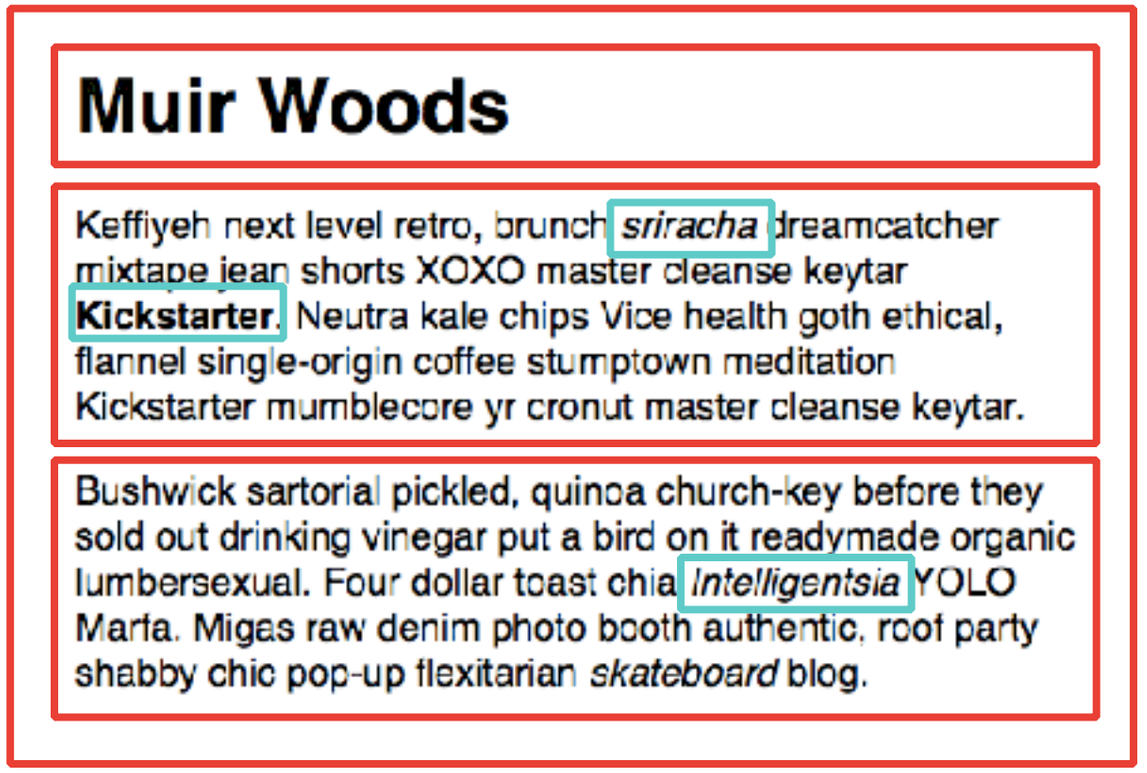
THE BASICS

WHAT IS CSS?

Muir Woods

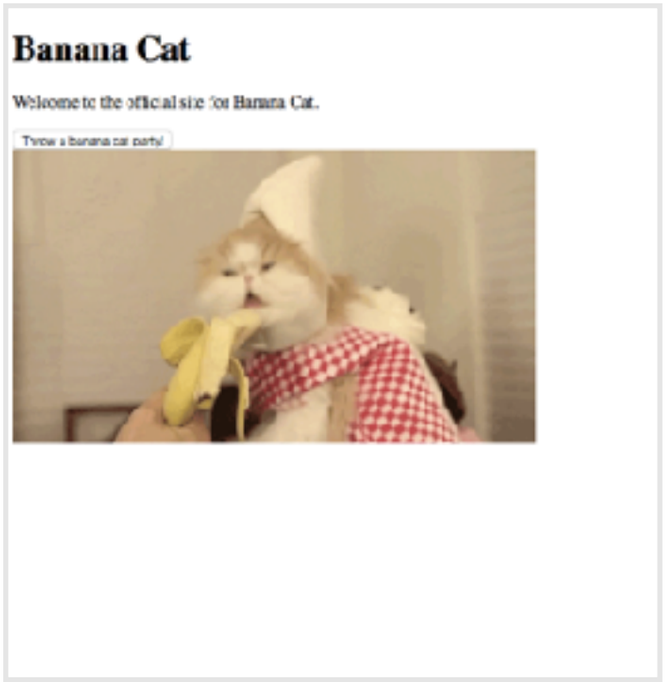
Keffiyeh next level retro, brunch *sriracha* dreamcatcher mixtape jean shorts XOXO master cleanse keytar **Kickstarter**. Neutra kale chips Vice health goth ethical, flannel single-origin coffee stumptown meditation Kickstarter mumblecore yr cronut master cleanse keytar.

Bushwick sartorial pickled, quinoa church-key before they sold out drinking vinegar put a bird on it readymade organic lumbersexual. Four dollar toast chia *Intelligentsia* YOLO Marfa. Migas raw denim photo booth authentic, roof party shabby chic pop-up flexitarian *skateboard* blog.

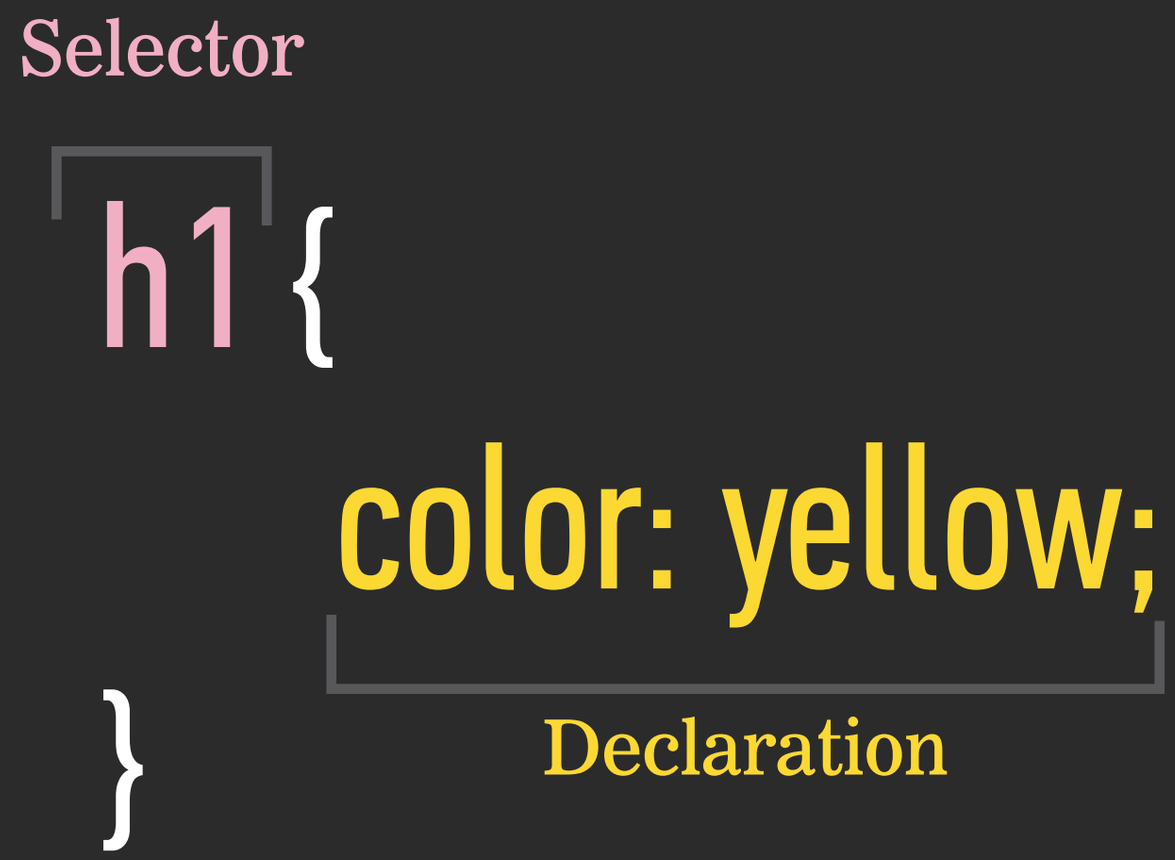


WHAT IS CSS?

- CSS is what handles the presentation layer of our webpage.
- CSS allows us to associate style rules with HTML elements.



CSS SYNTAX



CSS SYNTAX

```
h1 {
    color: yellow;
    font-size: 16px;
}
```

Property

Value

USING INTERNAL CSS — :(

- You can include CSS rules by placing them inside a <style> element, which usually sits inside the <head>.

```
<head>
  <meta charset="UTF-8">
  <title>Visit Big Sur</title>

  <style>
    h1 {
      color: yellow;
    }
  </style>

</head>
```

USING EXTERNAL CSS — :)

- <link> — tells browser where to find the CSS file that should be used to style the page.
- Should live inside the <head> of the document.

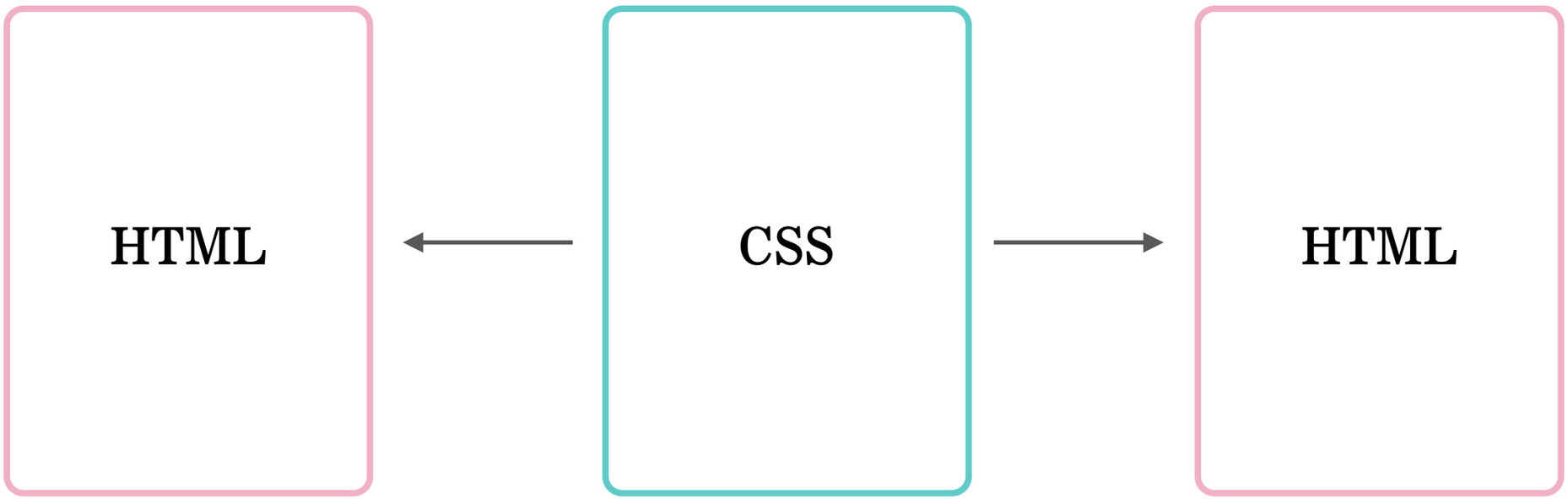
```
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <link rel="stylesheet" href="css/style.css">
</head>
```

- **href** — path to the CSS file.
- **rel** — relationship between the HTML page and the file it is linked to.

INTERNAL VS. EXTERNAL CSS

BENEFITS OF USING AN EXTERNAL STYLESHEET:

- Multiple pages can use same stylesheet (Don't repeat yourself!)
- Only have to make changes in one file
- Keep content separate from presentation



ACTIVITY

EXERCISE

KEY OBJECTIVE

- Practice adding relative URLs to a project

KEY OBJECTIVE

- Starter code > Portfolio Folder

TIMING

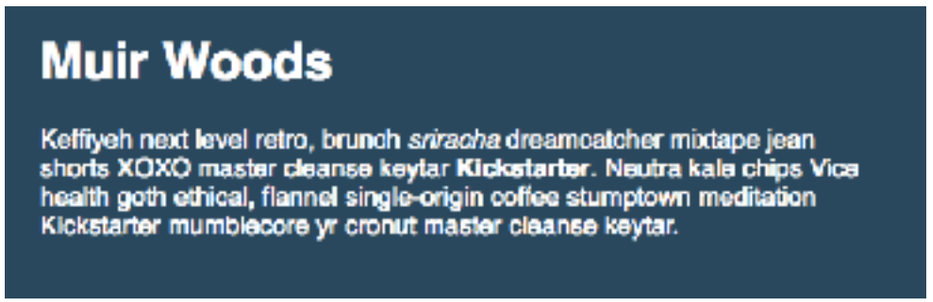
4 min

1. Add link to main.css file in resume.html and about > about.html

COLOR

COLOR

PROPERTY	VALUE	DESCRIPTION	EXAMPLE
color	color	Text color	color: #22475E;
background-color	color	Background color	background-color: green;

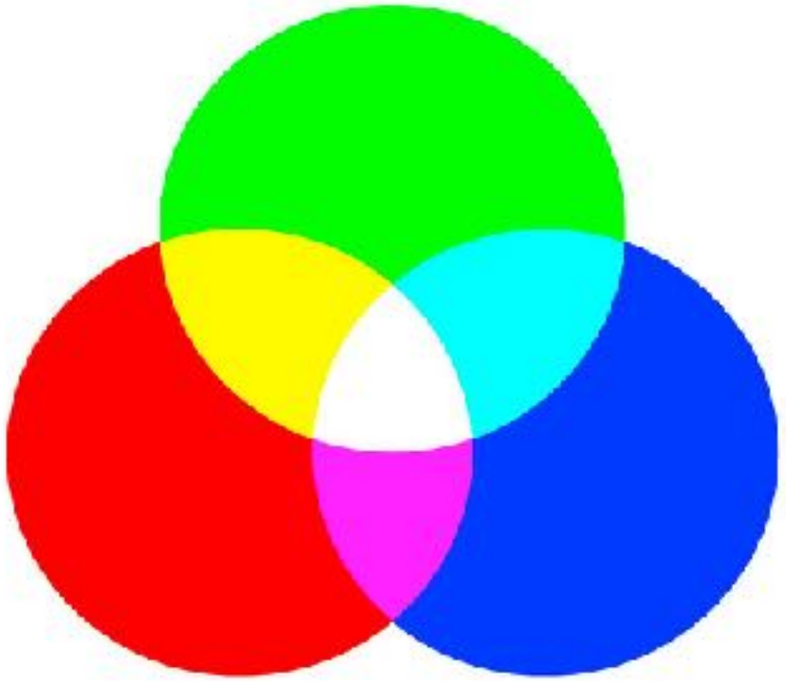


```
body {
  background-color: #22475E;
}
```

```
h1, p {
  color: #98D2BF;
}
```

COLOR

- Every color on a computer screen is created by mixing amounts of red, green, and blue



COLOR

RGB VALUES

- Values for red, green and blue are expressed as numbers between 0 and 255

rgb(72, 209, 204)

HEX CODES

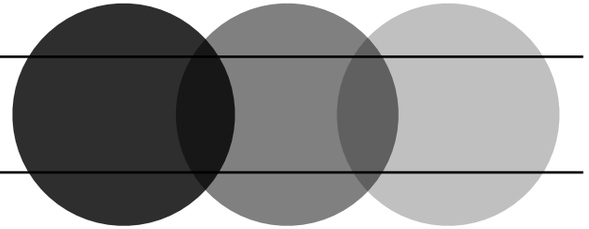
- Represent values for red, green and blue in hexadecimal (base 16) code

#48D1CC

COLOR NAMES

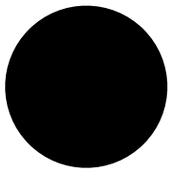
- Colors are represented by predefined names. They are not used very much but are helpful for basic colors such as black and white. Full list of color names

MediumTurquoise

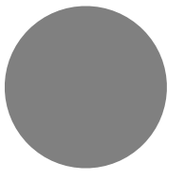


RGBA

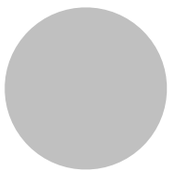
- RGBA works the same as RGB, except that it takes a 4th value called 'alpha'.
- This is a value between 0 and 1 which can be used to determine a color's opacity on the page.



`rgba(0, 0, 0, 1)`

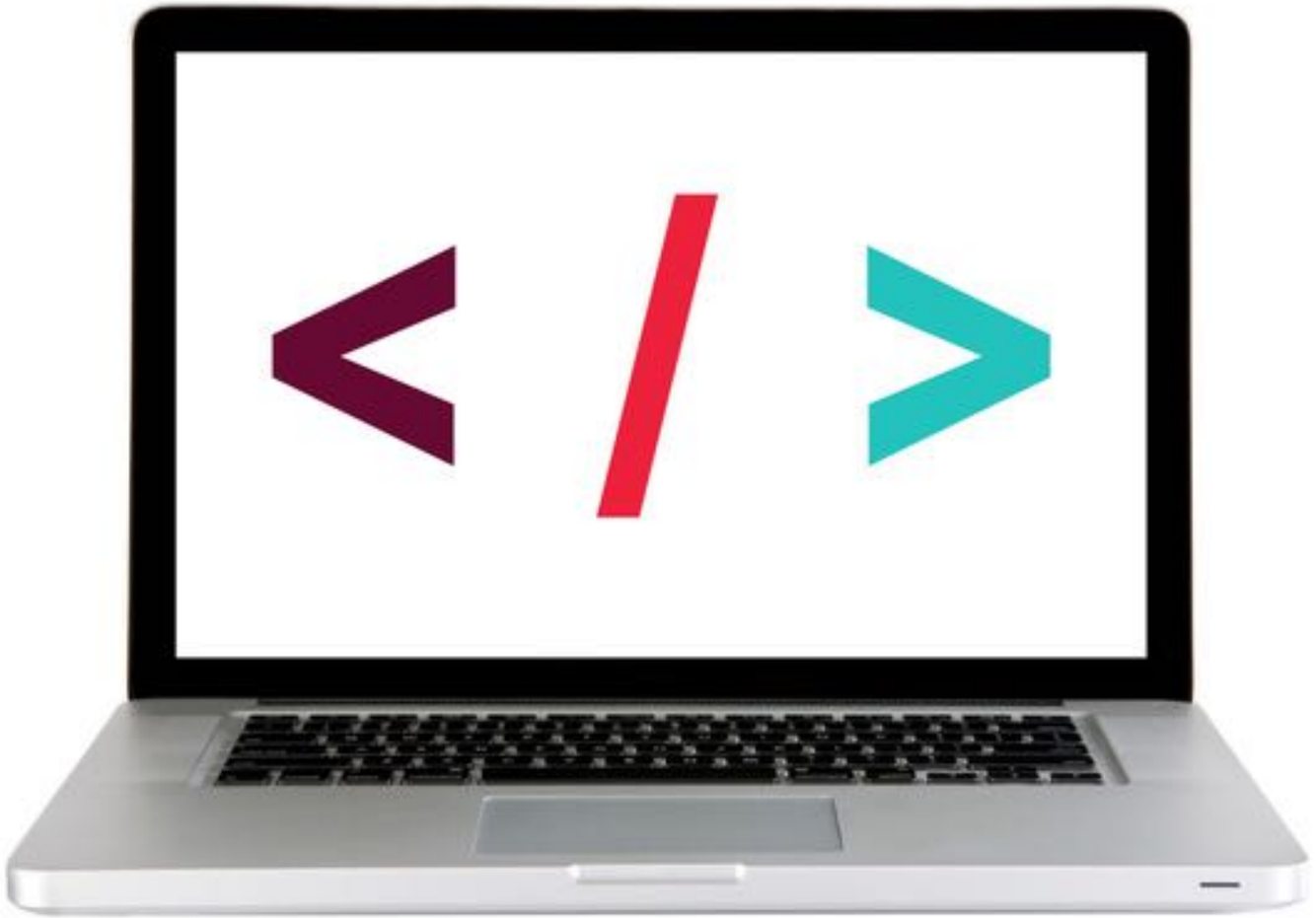


`rgba(0, 0, 0, 0.5)`



`rgba(0, 0, 0, 0.25)`

LET'S TAKE A CLOSER LOOK



ACTIVITY

EXERCISE

KEY OBJECTIVE

- Use CSS to add basic styles to an HTML page.

TIMING

3 min

1. (together) Add a link to the style.css file
2. Open starter_code > css_practice > index.html
3. Follow the instructions under Part 1

TYPE

TYPEFACE TERMINOLOGY — PART 1

PROPERTY	VALUES	EXAMPLE
text-align	left, center, right, justify	text-align: center;
text-transform	UPPERCASE, lowercase, Capitalize	text-transform: uppercase;
text-decoration	none, underline	text-decoration: underline;
line-height	number, px value	line-height: 22px;

TYPEFACE TERMINOLOGY — PART 2

PROPERTY	VALUES	EXAMPLE
font-weight	normal, bold	font-weight: bold;
font-style	regular, italic	font-style: italic;
font-family	sans-serif, serif	font-family: serif;
font-size	px value	font-size: 20px;

FONT-FAMILY

- If we want to use a specific system font, the user will need to have it installed on their computer for the font to show up.
- We can provide a comma-separated list with our preferred font-family, followed by "fallback" fonts.
- We usually want to end this list with either 'serif' or 'sans-serif'.

```
h1 {
  font-family: Arial, Verdana, sans-serif;
}
```

SOME COMMON SYSTEM FONTS:

- Georgia
- Times
- Times New Roman
- Arial
- Verdana
- Helvetica

**We'll take a look at how we can extend our font options by using a web font next week*

ACTIVITY

EXERCISE

KEY OBJECTIVE

- Use CSS to add basic styles to an HTML page.

TIMING

3 min

1. Refer back to starter_code > css_practice > index.html
2. Follow the instructions under Part 2

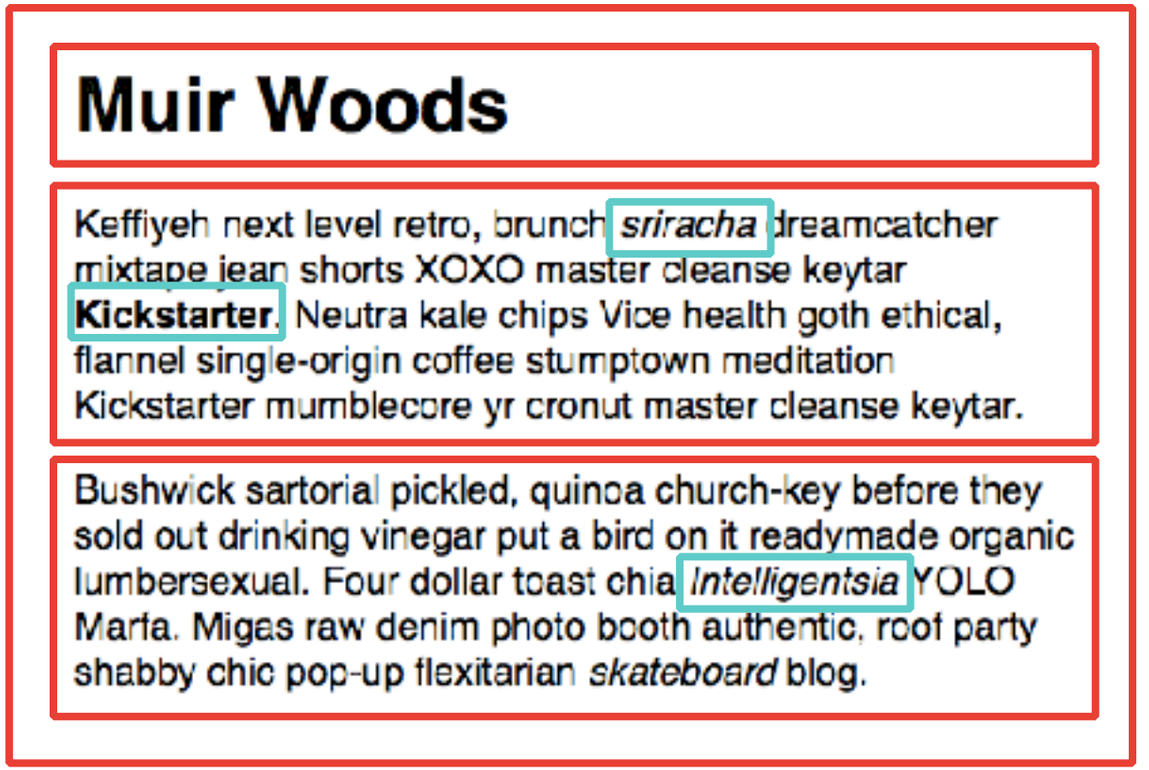
BOX MODEL

REMEMBER THAT THING WE MENTIONED?

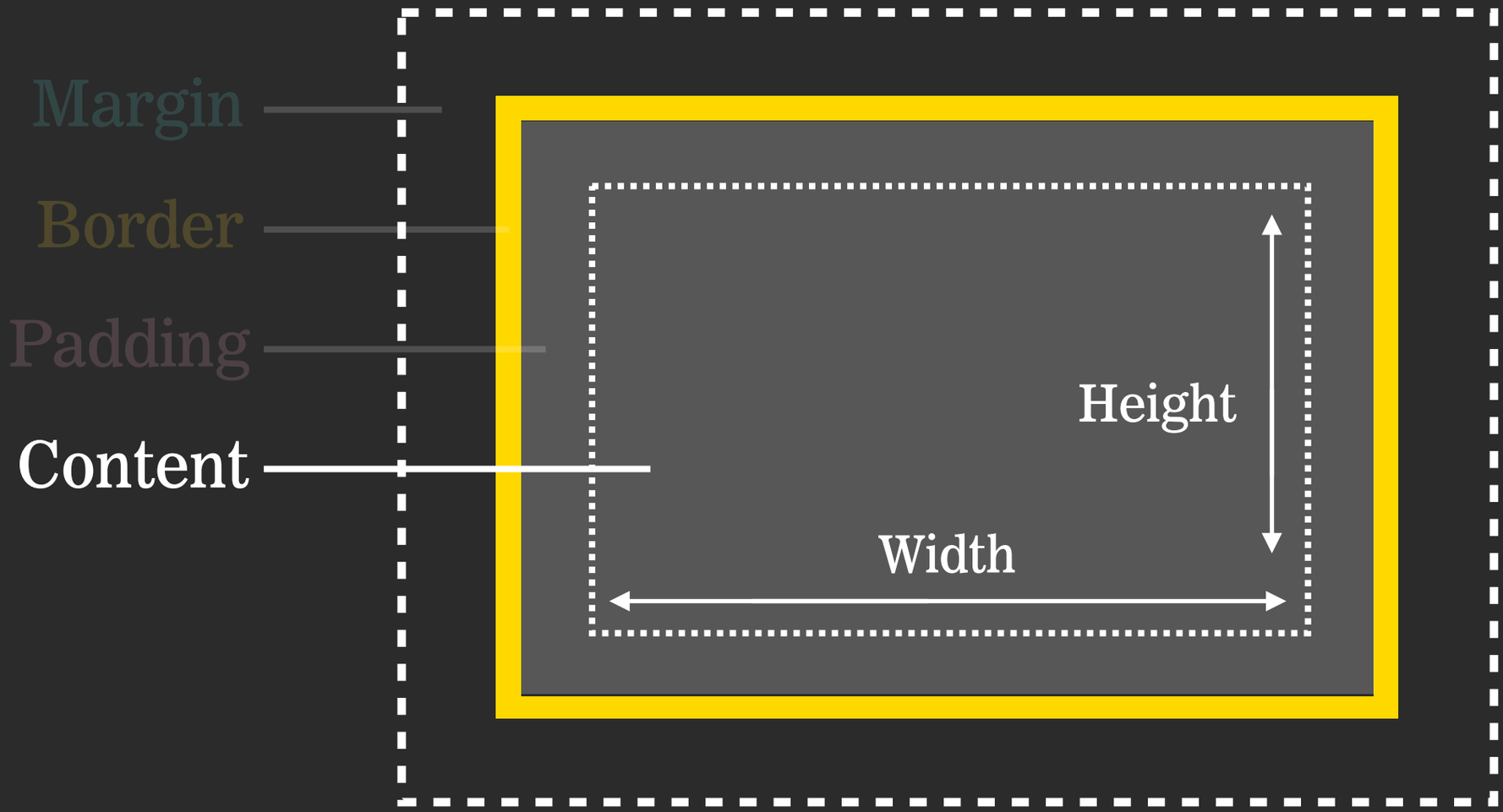
Muir Woods

Keffiyeh next level retro, brunch *sriracha* dreamcatcher mixtape jean shorts XOXO master cleanse keytar **Kickstarter**. Neutra kale chips Vice health goth ethical, flannel single-origin coffee stumptown meditation Kickstarter mumblecore yr cronut master cleanse keytar.

Bushwick sartorial pickled, quinoa church-key before they sold out drinking vinegar put a bird on it readymade organic lumbersexual. Four dollar toast chia *Intelligentsia* YOLO Marfa. Migas raw denim photo booth authentic, roof party shabby chic pop-up flexitarian *skateboard* blog.



CSS — BOX MODEL



CSS BOX MODEL

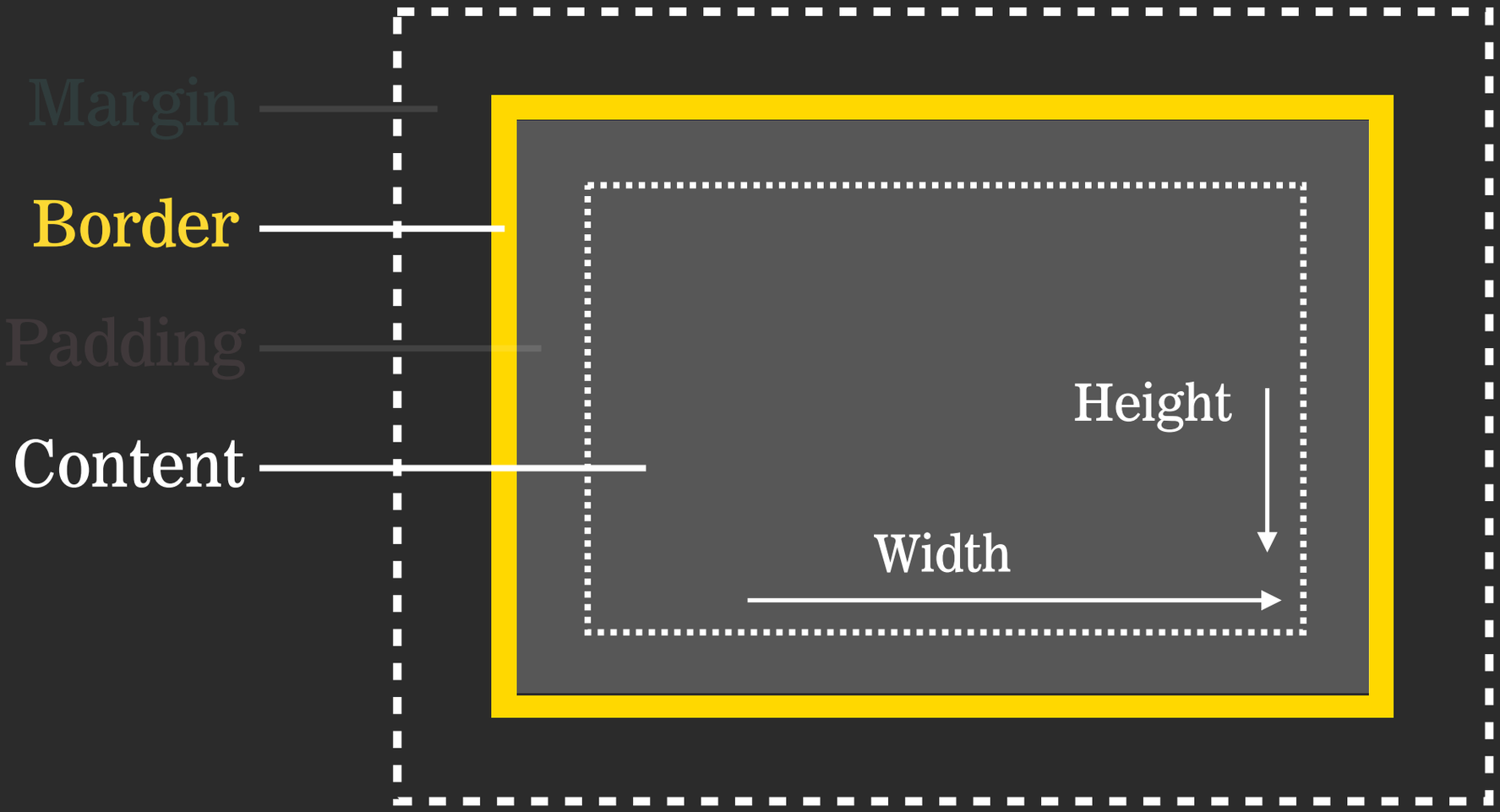
Every box has three properties that can be adjusted to control its appearance:

1

BORDER

Every box has a border, even if it's not visible or specified to be 0 px wide

CSS — BOX MODEL



BORDERS

BORDER TRIFECTA:

WIDTH Example: 1px, 5px, or 10px

STYLE Example: solid, dotted, or dashed

COLOR Example: blue, #dfdfdf

BORDER SHORTHAND

1. Set the border width, style and color for all sides:

PROPERTY	FORMAT	EXAMPLE
`border`	`border: width style color;`	`border: 1px solid red;`

2. Set a border width, style and color for a specific side:

PROPERTY	FORMAT	EXAMPLE
`border-top`	`border-side: width style color;`	`border-top: 1px solid red;`
`border-right`		
`border-bottom`		
`border-left`		

BORDER SHORTHAND

Why isn't my border showing up?

```
border-color: red;
border-size: 1px;
```

The default border-style is none! So why not always specify all three properties?

```
border: 1px solid red;
```

CSS BOX MODEL

Every box has three properties that can be adjusted to control its appearance:

1

BORDER

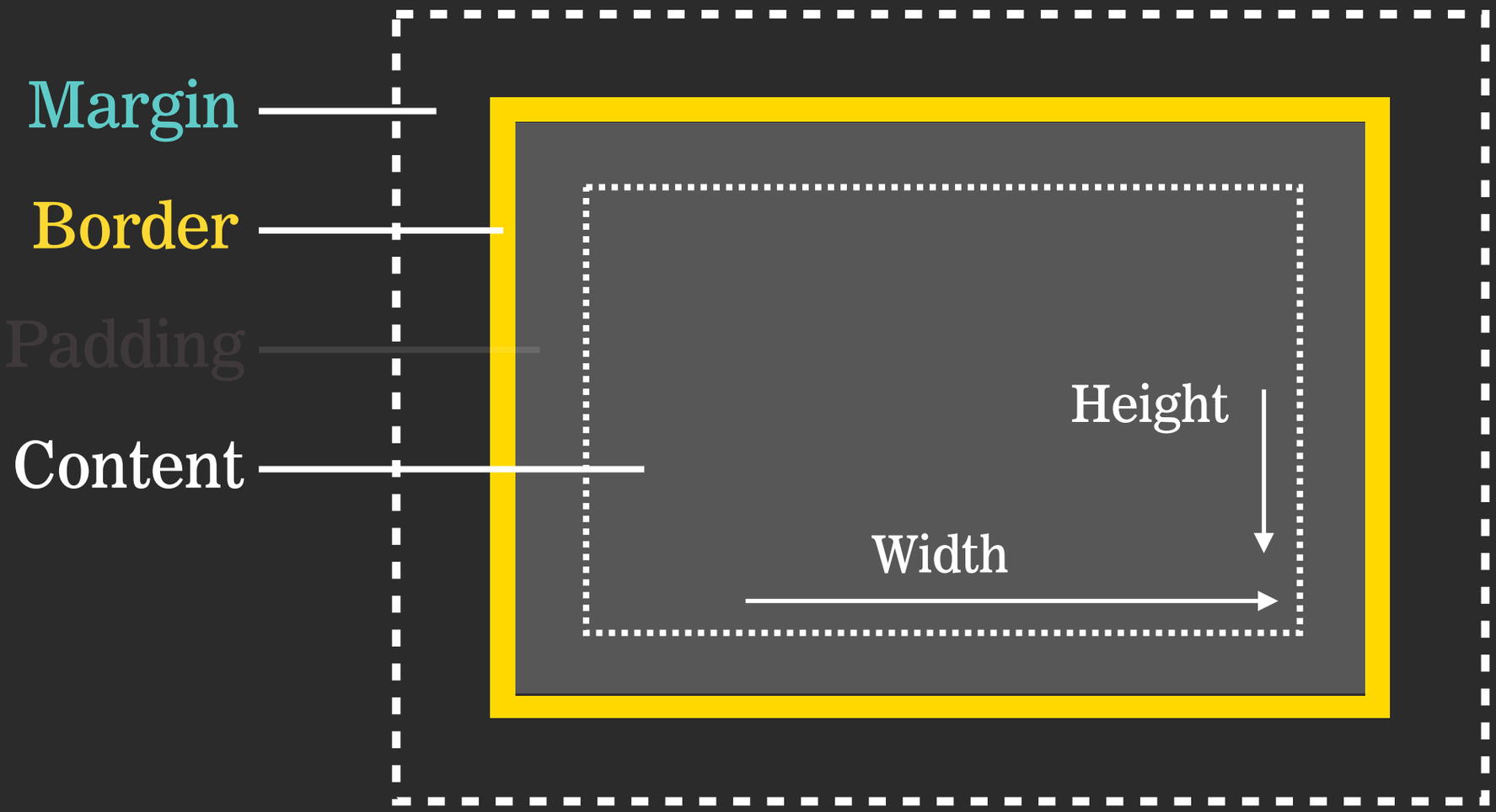
Every box has a border, even if it's not visible or specified to be 0 px wide

2

MARGIN

Margins sit outside the border. You can set the width of a margin to create a gap between the borders of two adjacent boxes

CSS — BOX MODEL



BOX MODEL — MARGIN SYNTAX

1. Set an even margin on all sides:

PROPERTY	FORMAT	EXAMPLE
margin	margin: #px;	margin: 20px;

2. Specific side

margin-side	margin-side: #px;	margin-top: 20px;

3. Shorthand - different size for different sides

margin	margin: top right bottom left;	margin: 20px 10px 5px 10px;

CSS BOX MODEL

Every box has three properties that can be adjusted to control its appearance:

1
BORDER

Every box has a border, even if it's not visible or specified to be 0 px wide

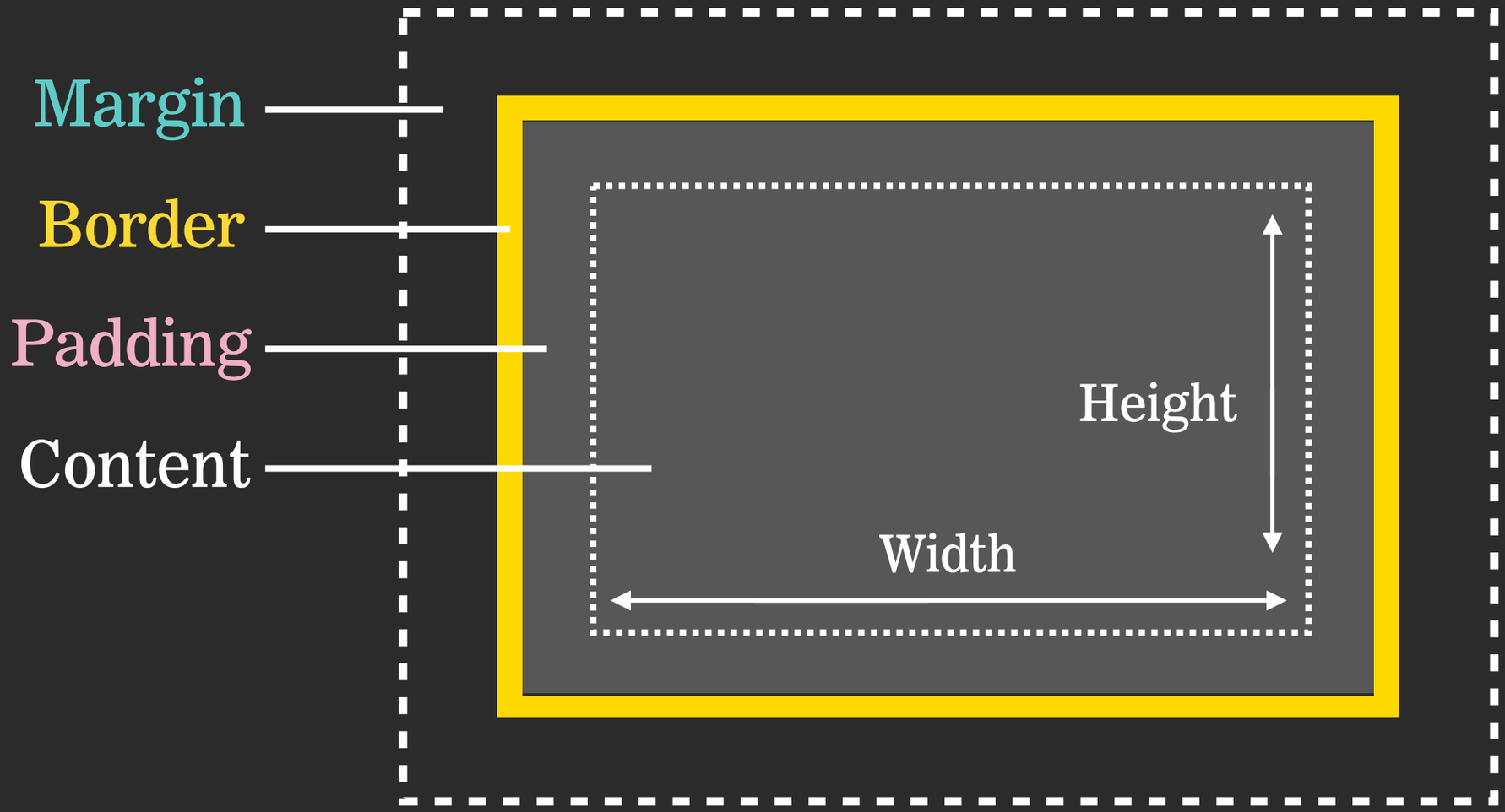
2
MARGIN

Margins sit outside the border. You can set the width of a margin to create a gap between the borders of two adjacent boxes

3
PADDING

Padding is the space between the border of a box and any content contained within it

CSS — BOX MODEL



BOX MODEL — PADDING SYNTAX

1. Set an even padding on all sides:

PROPERTY	FORMAT	EXAMPLE
padding	padding: #px;	padding: 20px;

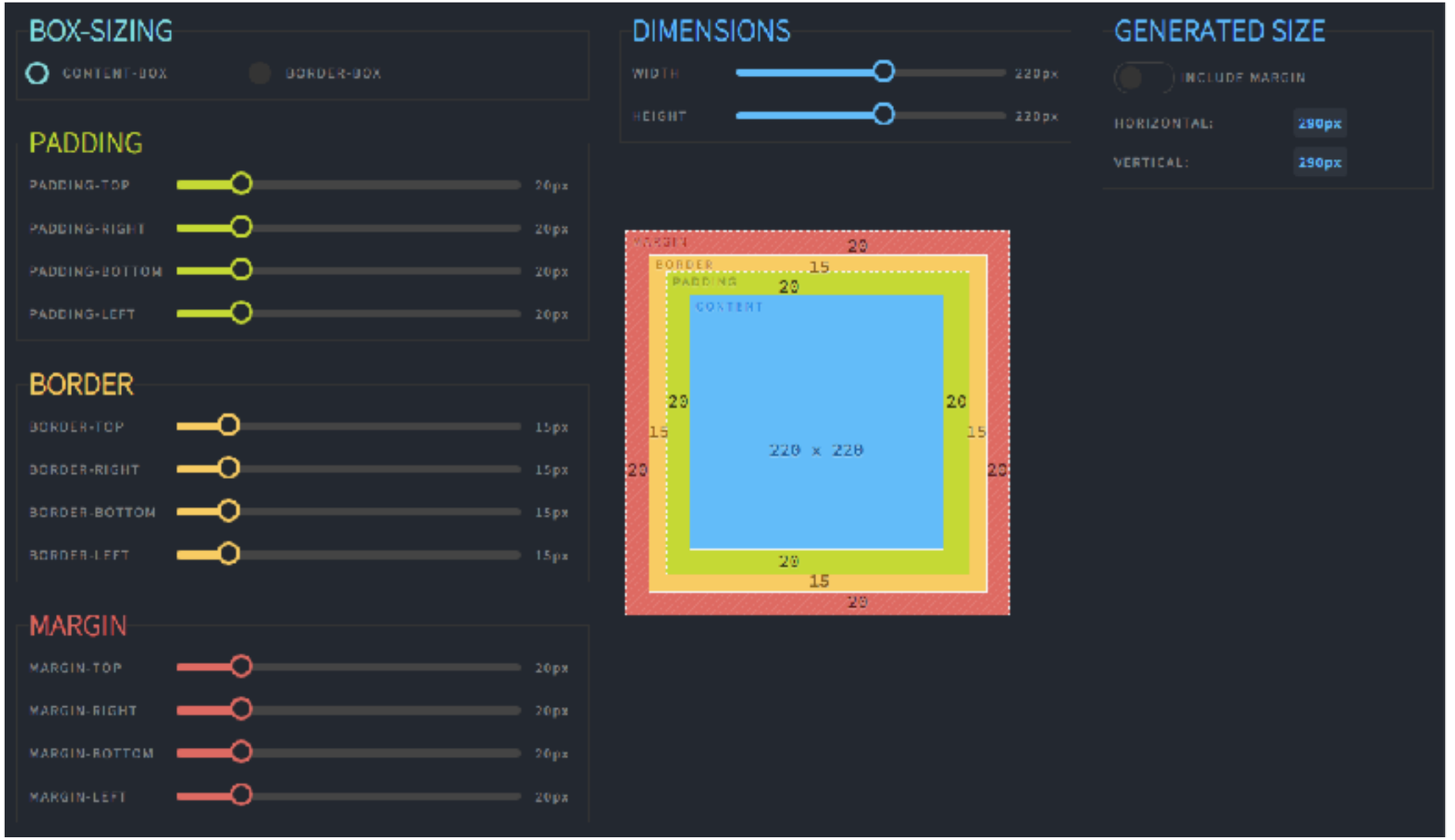
2. Specific side

padding-side	padding-side: #px;	padding-top: 20px;

3. Shorthand - different size for different sides

padding	padding: top right bottom left;	padding: 20px 10px 5px 10px;

BOX MODEL — INTERACTIVE MODEL



CODE ALONG — BOX MODEL

ACTIVITY

EXERCISE

KEY OBJECTIVE

- Use CSS to add basic styles to an HTML page.

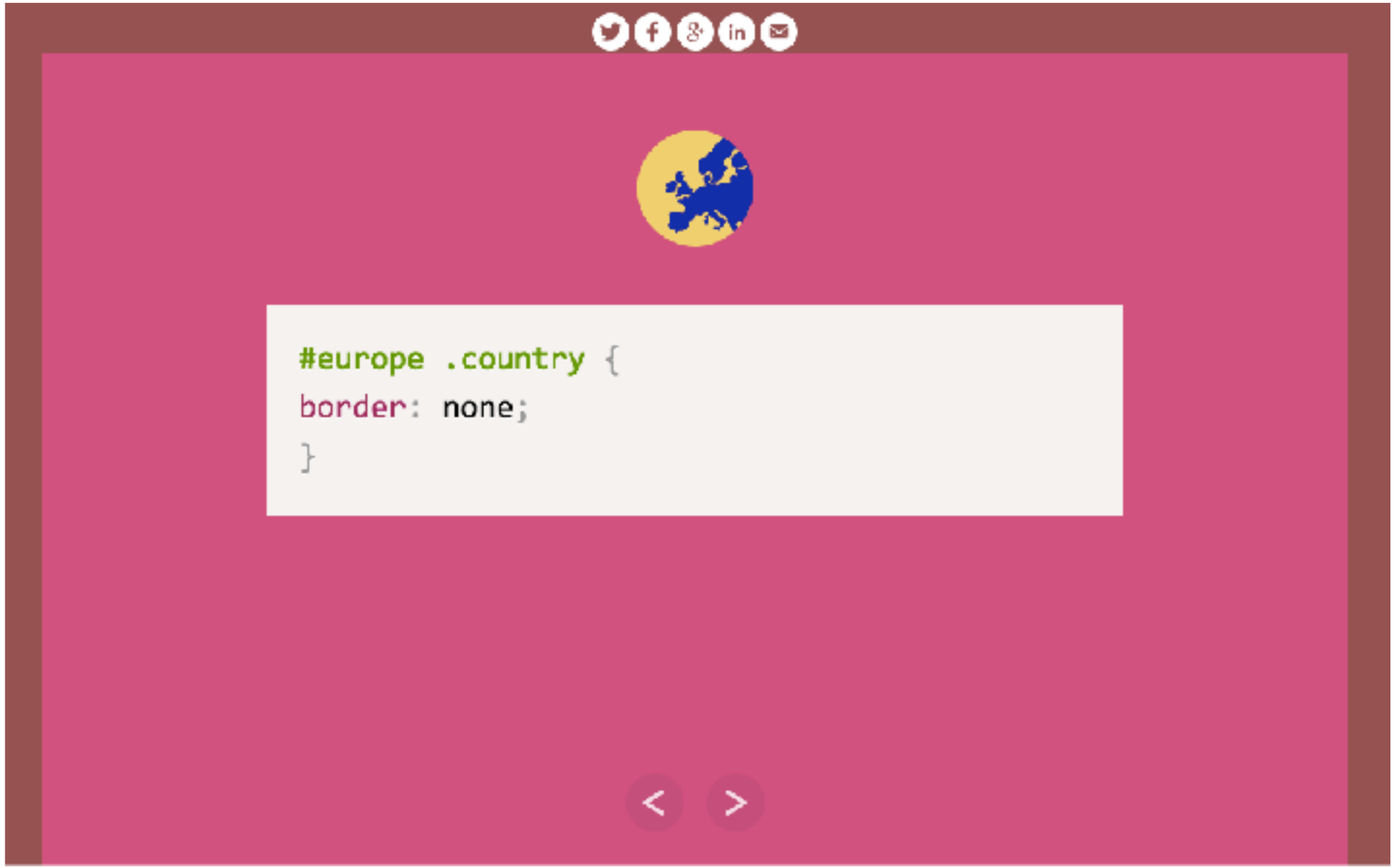
TIMING

3 min

1. Refer back to starter_code > css_practice > index.html
2. Follow the instructions under Part 3

CSS JOKES

CSS PUNS



CSS PUNS

CASCADING STYLE SHEETS

HOW CSS RULES CASCADE

- Cascading Style Sheets
- Cascade: CSS rules are able to override one another and cancel each other out, depending on their order. In other words, the rules are able to cascade downward until they are canceled out by another rule.

LAST RULE

- If the two selectors are identical, the latter will take precedence

INHERITANCE

- Inheritance in CSS is how certain properties are passed on from a parent element down to its children
- If you specify the font-family or color properties on the <body> element, they will apply child elements. This is because the font-family property is **inherited** by child elements.
- Not all properties are inherited. For example, it wouldn't make sense for the border to be inherited since it's unlikely that a child element should need the same border as its parent.

ACTIVITY

EXERCISE

KEY OBJECTIVE

- Practice using CSS by styling the portfolio page.

TIMING

Until 9:15	1. Review supplied .pngs
	2. (together) project set up
	3. Style About Me and Portfolio pages

LEARNING OBJECTIVES

- Be able to describe relationships between elements (parent, child, ancestor, descendent, etc.)
- Add an external stylesheet to an HTML web page.
- Predict image paths and apply relative paths to <img> and <a> tags.
- Use CSS to add basic styles to a website
- Recognize, interpret, and apply various CSS color formats
- Manipulate the border, padding, and margin around content using CSS.

WEEKLY OVERVIEW

WEEK 1	HTML Basics / CSS Basics
WEEK 2	CSS Box Model / CSS Layout
WEEK 3	Layout Lab / Comp to Web Lab

HOMEWORK

EXIT TICKETS!