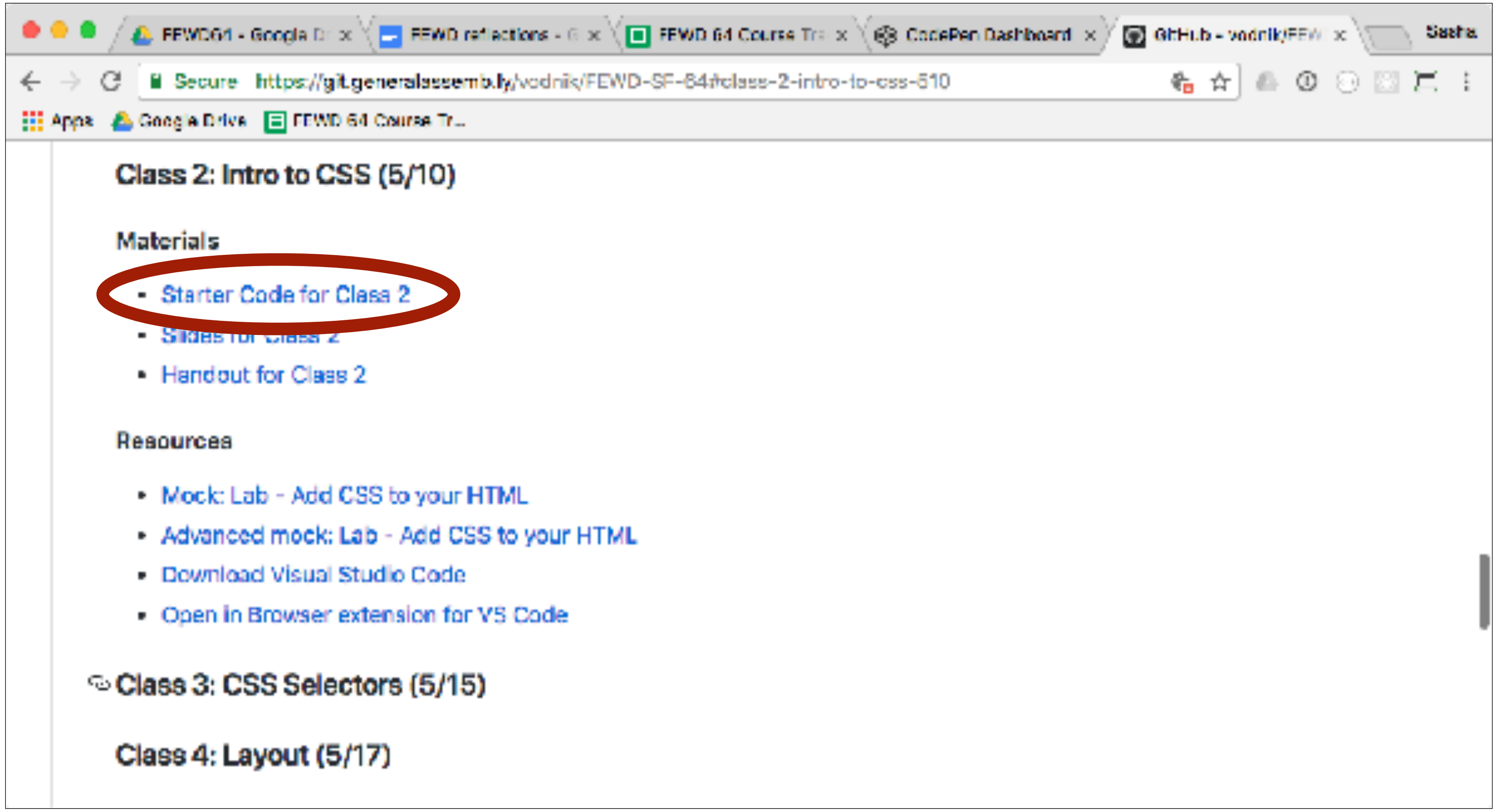
HELLO!

- Go to the class repo on GitHub Enterprise
- Click the **Starter Code for Class 3** link to download the files
- Move the downloaded file to your FEWD folder
- Unzip the files



Class 2: Intro to CSS (5/10)

Materials

- Starter Code for Class 2
- Slides for Class 2
- Handout for Class 2

Resources

- Mock: Lab - Add CSS to your HTML
- Advanced mock: Lab - Add CSS to your HTML
- Download Visual Studio Code
- Open in Browser extension for VS Code

Class 3: CSS Selectors (5/15)

Class 4: Layout (5/17)

WEEKLY OVERVIEW

WEEK 2	CSS Selectors / Layout
WEEK 3	Layout Lab / Grid Systems & Wireframing
WEEK 4	Responsive Design / CSS Positioning

LEARNING OBJECTIVES

- Add classes and IDs to HTML elements and apply CSS styles to elements based on class and ID.
- Explain when you would use a class and when you would use an ID.
- Apply CSS to elements based on their relationships.
- Describe inheritance in CSS.

AGENDA

Review

Box Model

CSS Selectors

Inheritance

Lab Time

EXIT TICKET QUESTIONS

1. Still a little confused about the HTML 5 elements and how they are "different" from things like <p>, <h1>, etc.
2. I want a little more clarification on when we would use main, section and article.
3. suggestion: breaking down cool websites, such as medium, facebook…

EXAMPLE: WHAT DOES SUCCESS LOOK LIKE?

- “I’ll need to Google for the exact syntax, but I know there’s an attribute to add an external stylesheet to an HTML web page.”
- “I can add an image to my website. Sometimes I look in the wrong folder, but I figure it out later.”
- “I know there are HTML elements like header and footer to mark out website structure. I know there are some other ones, but I’ll have to look that up.”
- “I know there are CSS properties like border and padding that I can use to add more space around content. I haven’t used them much, but I know they exist.”
- “When the colors in my text editor are weird, I know something’s up.”

WHAT DO I DO WHEN I GET STUCK?

- Indent your code.
- Review the Cheat Sheets.
- Make educated guesses based on course material + cheat sheets.
- If your guess doesn't work, DELETE IT and try a different guess.
- Take a break and work on something else.
- If you're stuck for more than 30 minutes, ASK US (exit tickets + office hours).

HOMEWORK REVIEW

ACTIVITY

EXERCISE

KEY OBJECTIVE

- Review portfolio project and show off your work

TYPE OF EXERCISE

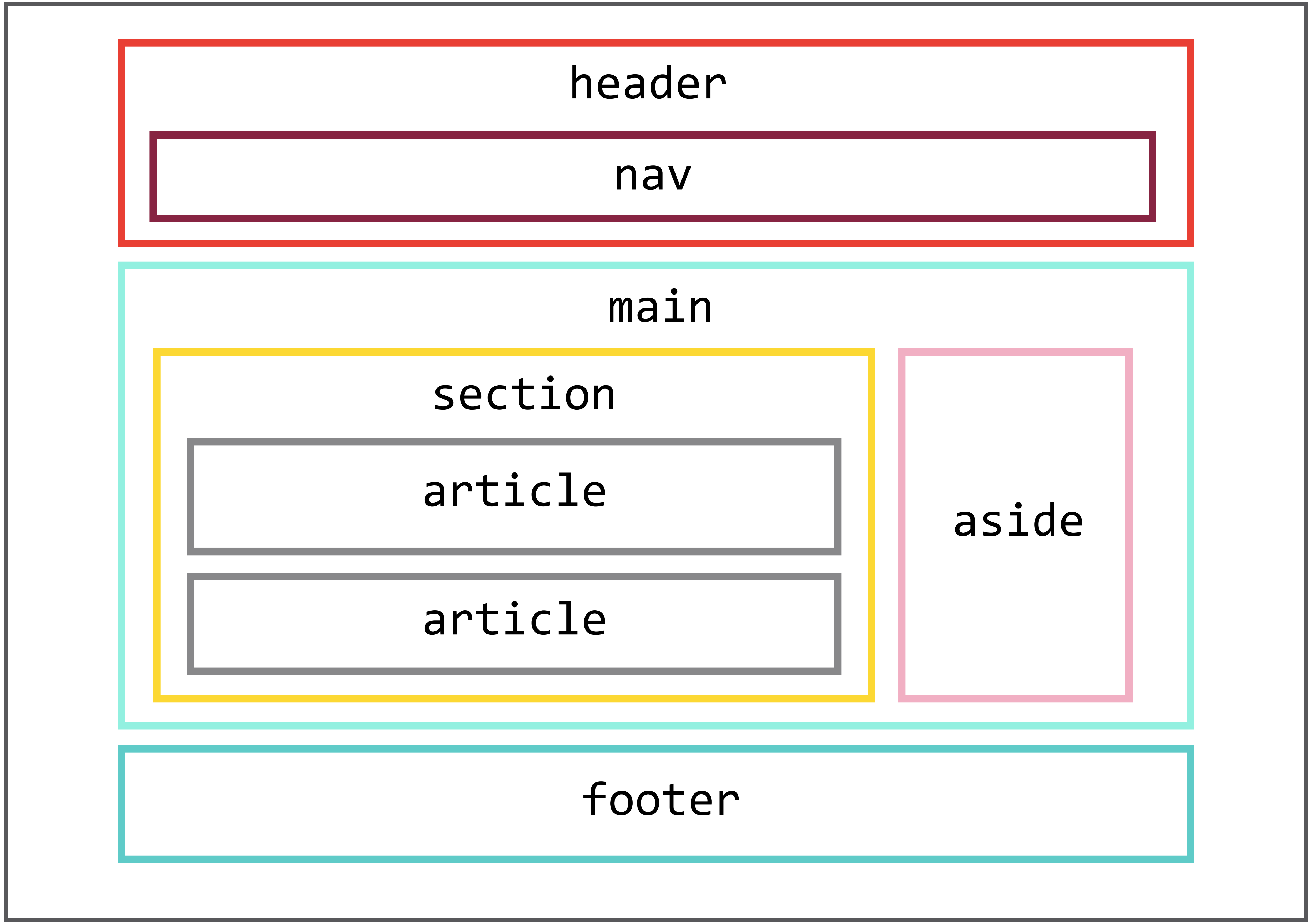
- Groups of 3-4

TIMING

6 min

1. Open portfolio sites on laptops and display them proudly!
2. Give feedback to your peers: "I like" and "I wish/wonder”
3. Share a challenge you ran into in your project and discuss how other group members may have worked with it.

HTML CONTINUED: STRUCTURAL ELEMENTS



header

nav

main

section

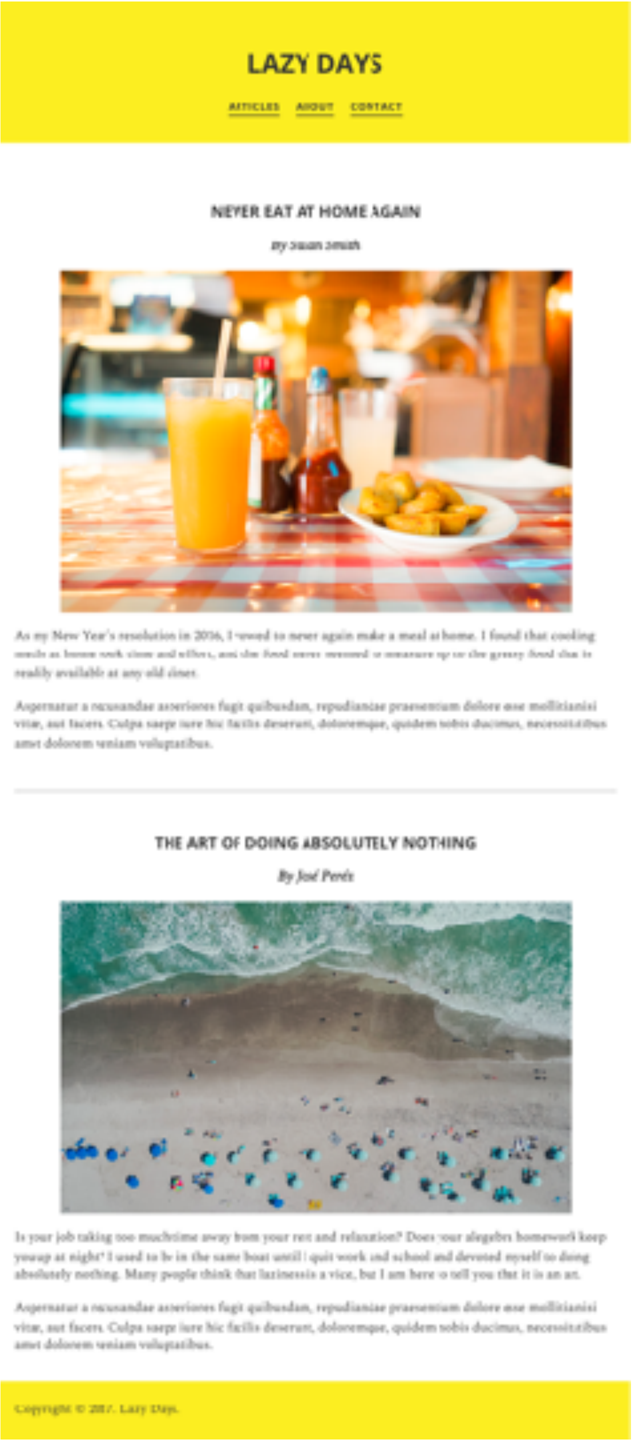
article

article

aside

footer

	DESCRIPTION
HEADER	Does it contain introductory content, such as the site title and navigation?
NAV	Does it contain a major navigational block?
FOOTER	Does it contain footer content?
MAIN	Does it contain content that is unique to that particular page?
ARTICLE	Is it self-contained, standalone content? Would it make sense on its own?
ASIDE	Is the content **not** required to understand the rest of the page?
SECTION	Is it a general section of the page that could have a heading?
DIV	Do you need a wrapper for styling and none of the above apply?



LAZY DAYS

ARTICLES ABOUT CONTACT

NEVER EAT AT HOME AGAIN

By Susan Smith

As my New Year's resolution in 2016, I vowed to never again make a meal at home. I found that cooking meals at home took time and effort, and the food never seemed to measure up to the greasy food that is readily available at any old diner.

Aspernatur a recusandae asperiores fugit quibusdam, repudiandae praesentium dolore esse mollitianisi vitae, aut facere. Culpa saepe iure hic facilis deserunt, doloremque, quidem nobis ducimus, necessitatibus amet dolorem veniam voluptatibus.

THE ART OF DOING ABSOLUTELY NOTHING

By José Peréz

Is your job taking too muchtime away from your rest and relaxation? Does your algebra homework keep youup at night? I used to be in the same boat until I quit work and school and devoted myself to doing absolutely nothing. Many people think that lazinessis a vice, but I am here to tell you that it is an art.

Aspernatur a recusandae asperiores fugit quibusdam, repudiandae praesentium dolore esse mollitianisi vitae, aut facere. Culpa saepe iure hic facilis deserunt, doloremque, quidem nobis ducimus, necessitatibus amet dolorem veniam voluptatibus.

Copyright © 2017. Lazy Days.

LAB: LAZY DAYS BLOG PART 1

EXERCISE

KEY OBJECTIVE

- Practice structuring a page with HTML5 elements.

TIMING

5 min

1. In Visual Studio Code, open the `lazy_days` folder in your starter code
2. Add HTML (including tags such as `header, main, article`) to the page using `part_1.png` as a guide
3. Style the page using `part_2.png` as a guide

CSS CONTINUED: BOX MODEL

READING: CSS BOX MODEL

Read more about the suburban neighborhood example

ACTIVITY: DISCUSS THE CSS BOX MODEL

EXERCISE

TYPE OF EXERCISE

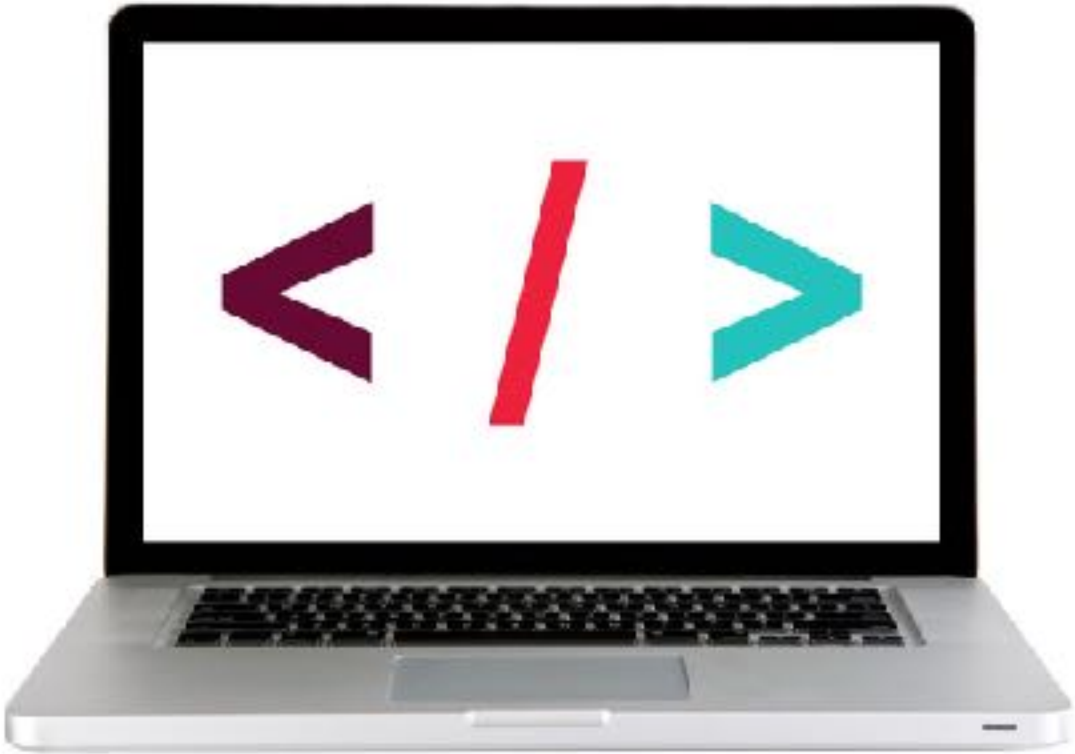
- Pairs

TIMING

5 min

1. Create a diagram that explains the CSS Box Model
2. Make sure to include margin, padding, border, content, height, and width
3. Prepare to share with the class!

DEMO: BOX MODEL BUTTON



https://codepen.io/svodnik/pen/pVLOMg

CONTENT

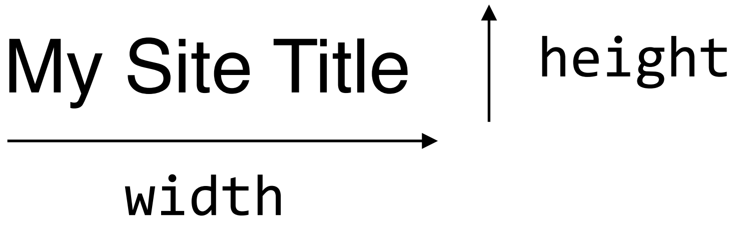
CONTENT

```
<h1>My Site Title</h1>
```

CONTENT

Content → My Site Title

CONTENT



My Site Title

height

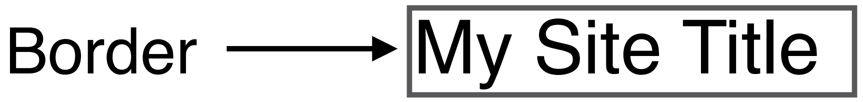
width

CONTENT

```
h1 {
  width: 300px;
  height: 40px;
}
```

BORDERS

BORDERS



Border

My Site Title

BORDER SHORTHAND

For a border on all sides:

```
border: width style color;          border: 1px solid red;
```

For a border on one side:

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

BORDER SHORTHAND

Why isn't my border showing up?

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

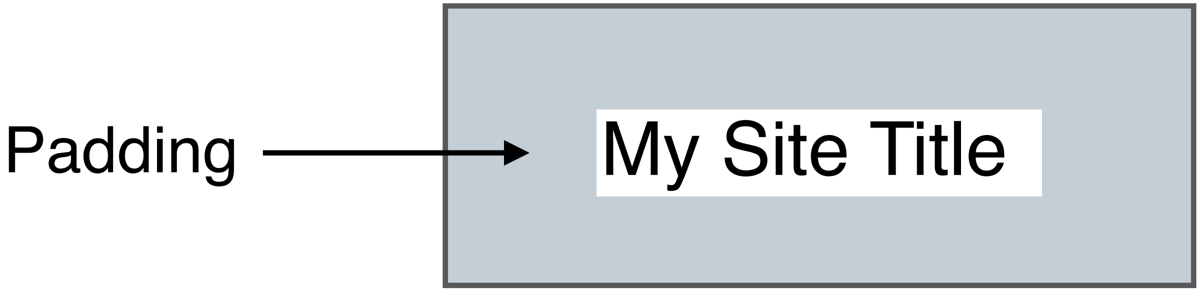
The default border-style is none! So why not always specify all three properties?

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

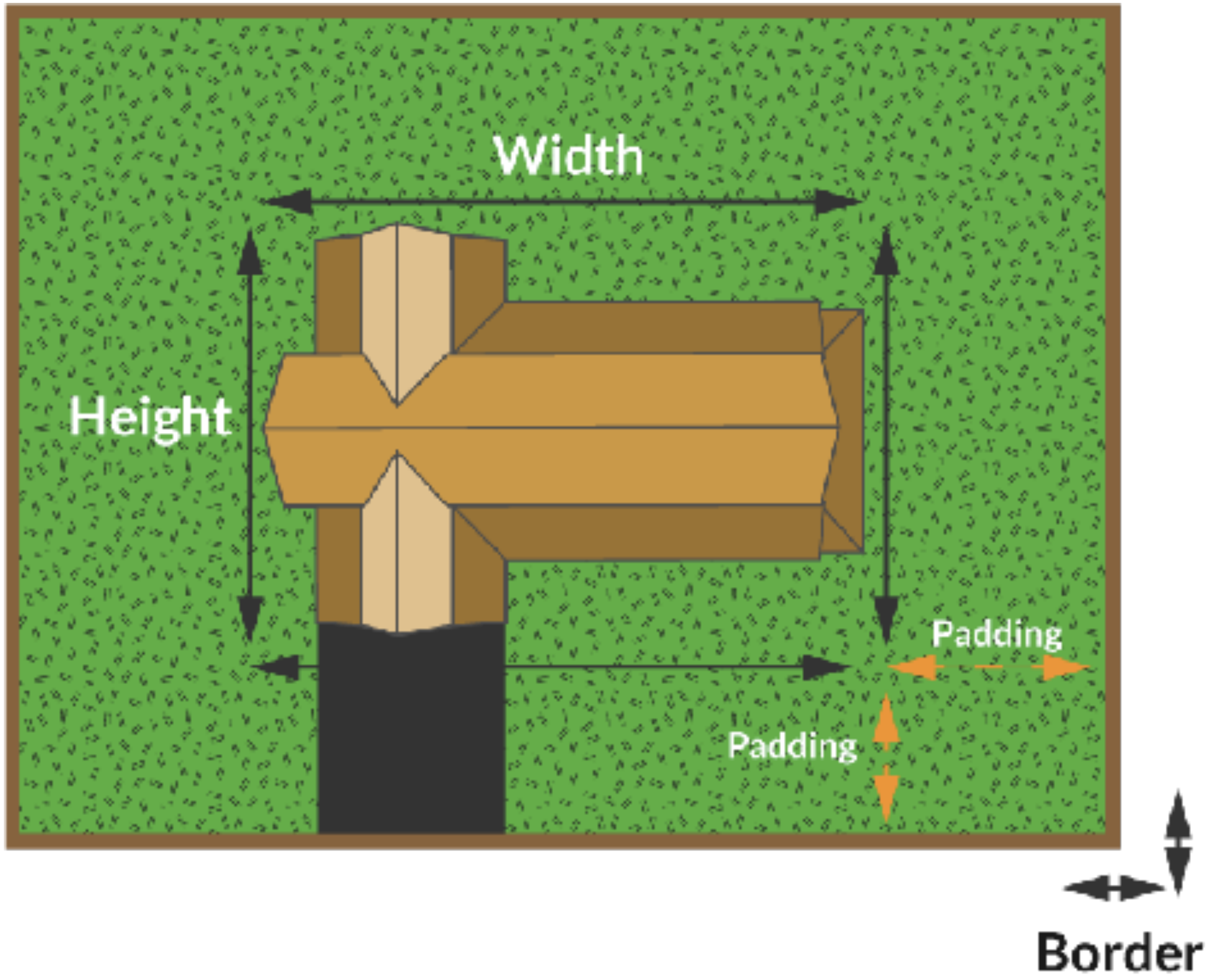
PADDING

PADDING

We can add `padding` to get space between our content and the border



PADDING



BOX MODEL — PADDING SYNTAX

For even padding on all sides:

```
padding: ##px;   padding: 20px;
```

For padding on one specific side:

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

For different size padding for each side:

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

Memory device: Trouble!

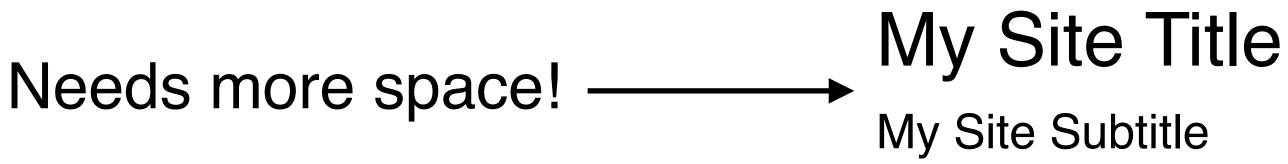
Shorthand if top/bottom are the same and left/right are the same:

```
padding: top/bottom right/left;   padding: 20px 10px;
```

MARGIN

MARGIN

We can add `margin` to add space between elements on our pages.

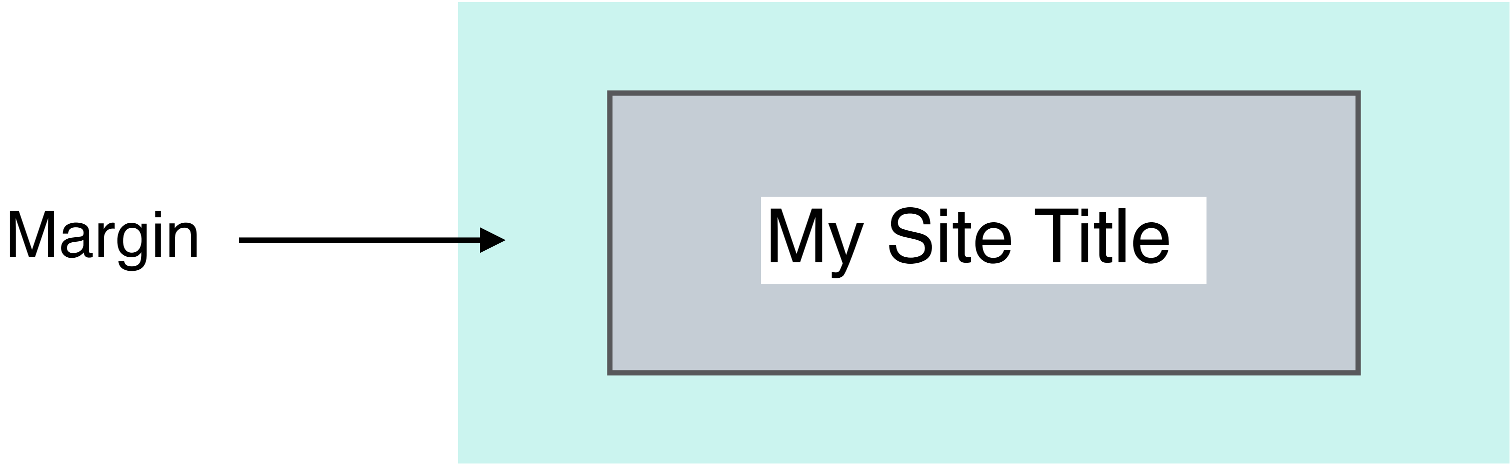


Needs more space!

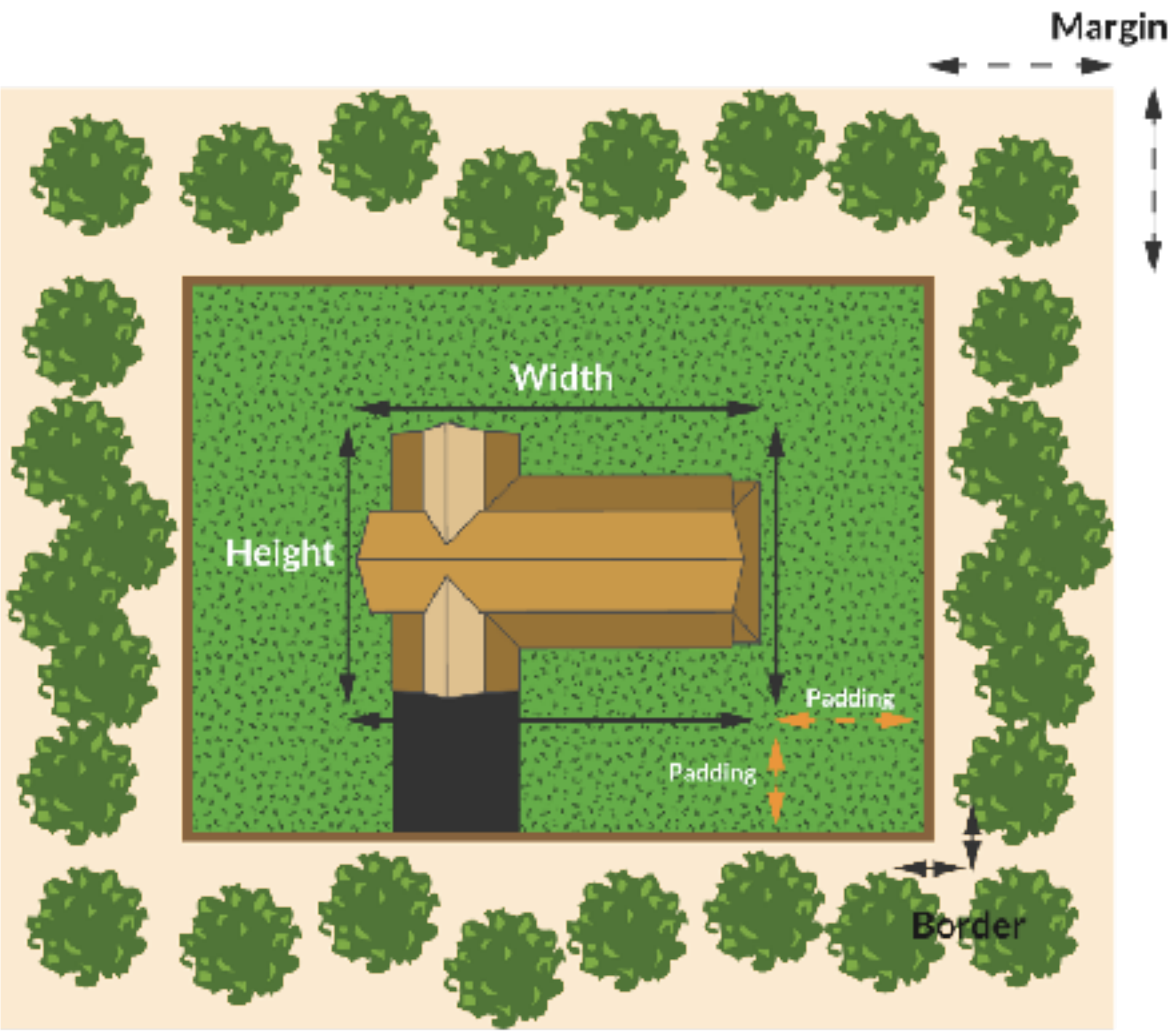
My Site Title

My Site Subtitle

MARGIN



MARGIN



BOX MODEL — PADDING SYNTAX

For even margin on all sides:

```
margin: ##px;    margin: 20px;
```

For margin on one specific side:

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

For different size margin for each side:

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

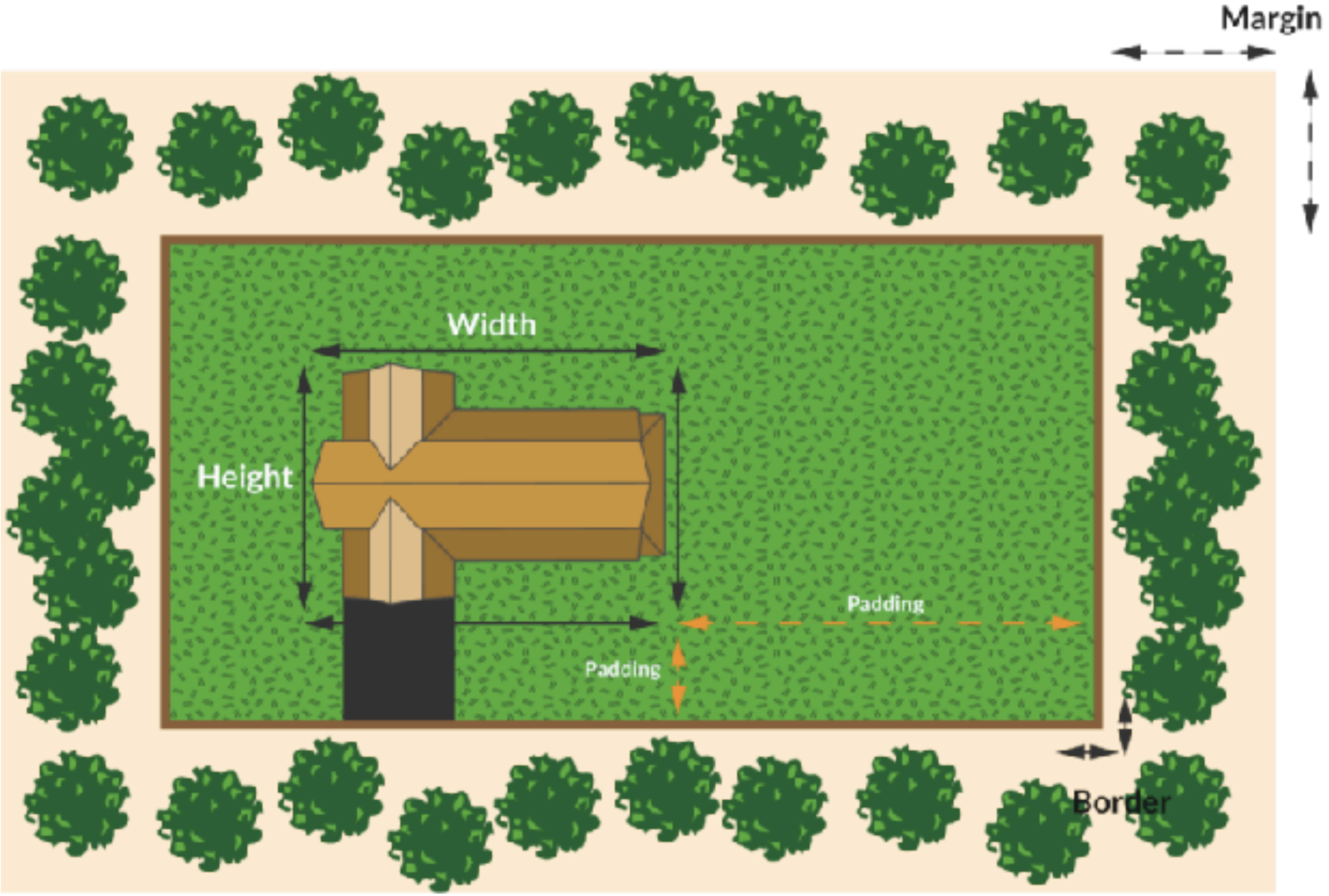
Memory device: Trouble!

Shorthand if top/bottom are the same and left/right are the same:

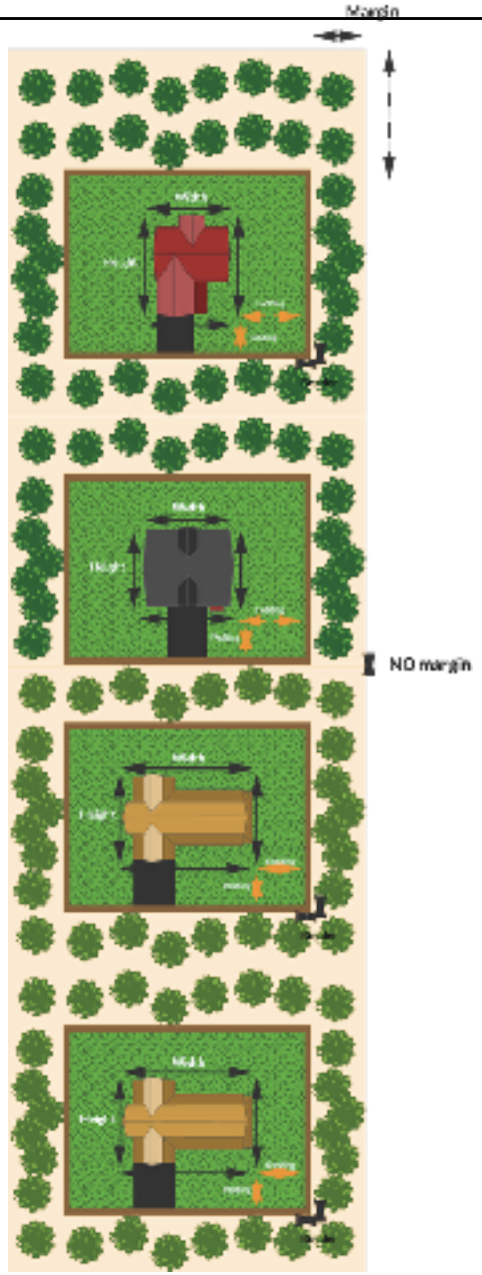
```
margin: top/bottom right/left;    margin: 20px 10px;
```

PADDING VS. MARGIN

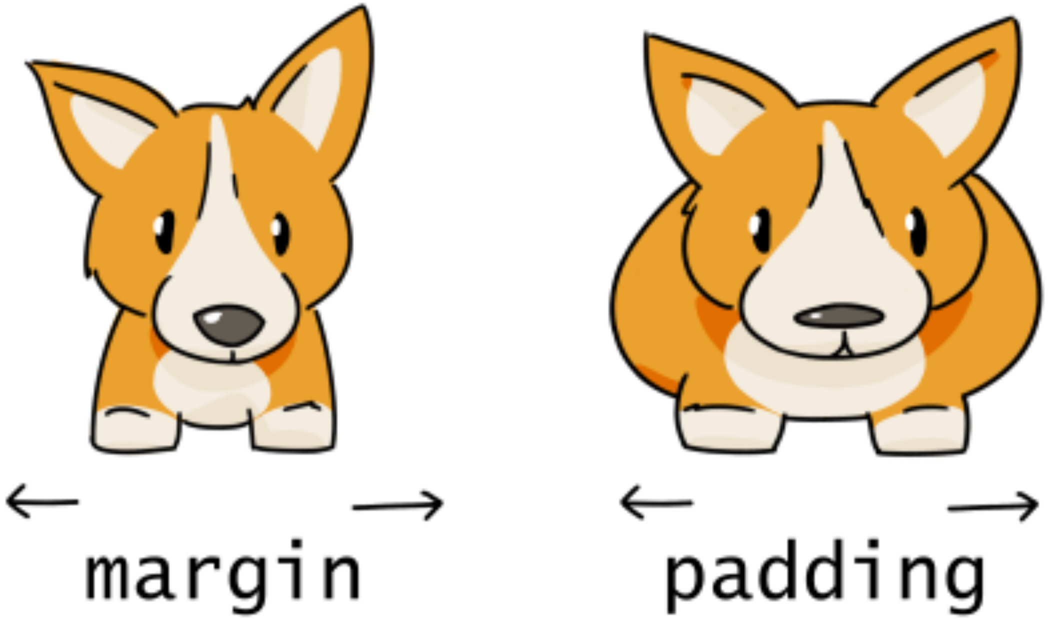
PADDING



MARGIN



CSS CORGIS!



LAB: LAZY DAYS BLOG PART 2

EXERCISE

KEY OBJECTIVE

- Manipulate the border, padding, and margin around content using CSS.

TIMING

5 min

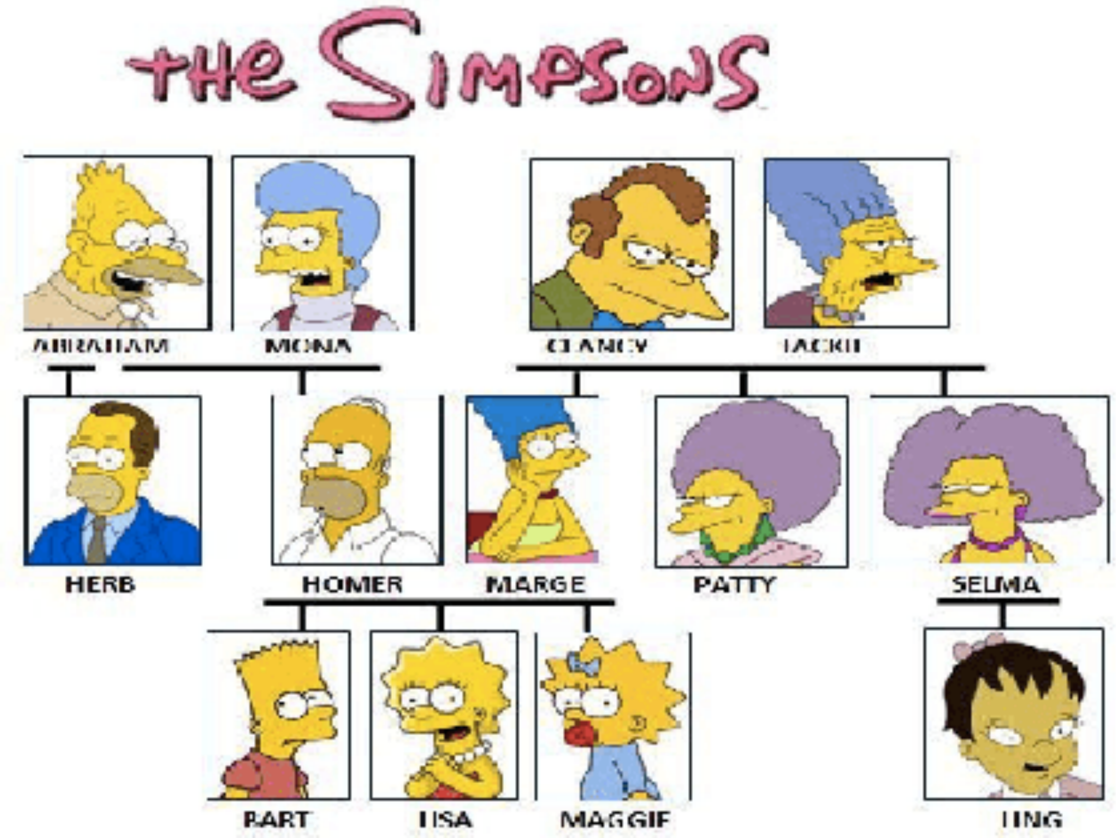
1. Open the “lazy_days” folder in your starter code
2. Practice adding padding, margin, and borders using part_3.png as a reference.
3. Bonus: Style the page as shown in the advanced image
4. Bonus #2: Make links change color on hover

CSS SELECTORS

HTML STRUCTURE

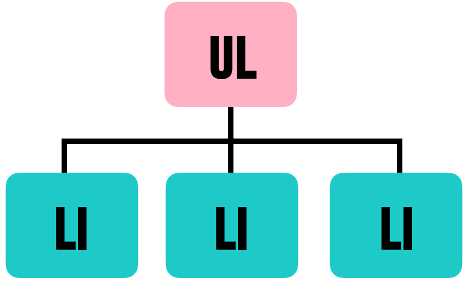
PAGE STRUCTURE — RELATIONSHIPS

- 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

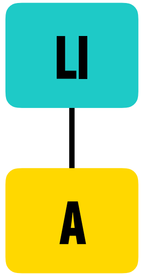
- Here we can say that our <ul> is the **parent** of our <li>s
- We can also say that our <li>s are **children** of the <ul>



```
<!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

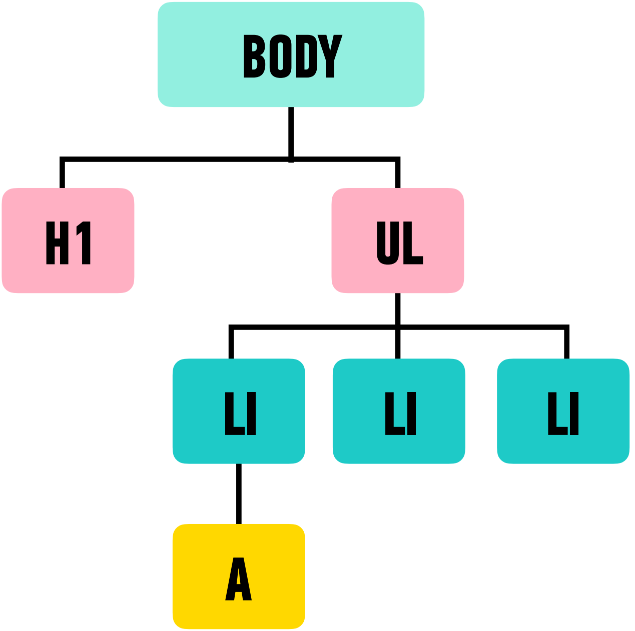
- Similarly, we can have an <a> tag that is nested inside, or wrapped by, our <li> element.



```
<!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>
```

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

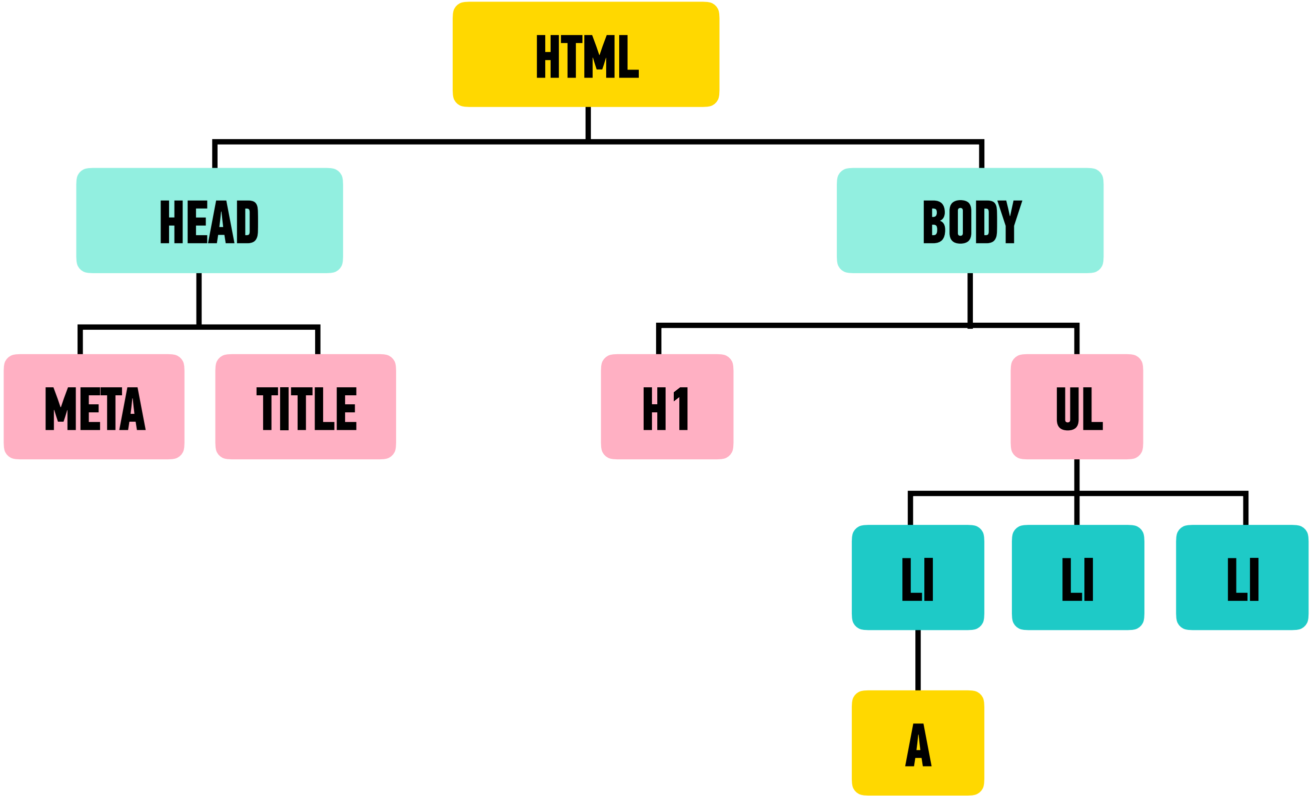


body

```
<!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 content

DOM ("DOCUMENT OBJECT MODEL") TREE



```html
<!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: DOM TREE

EXERCISE

KEY OBJECTIVE

- Be able to describe relationships between elements.

LOCATION

- Starter code > dom_tree
- Whiteboard

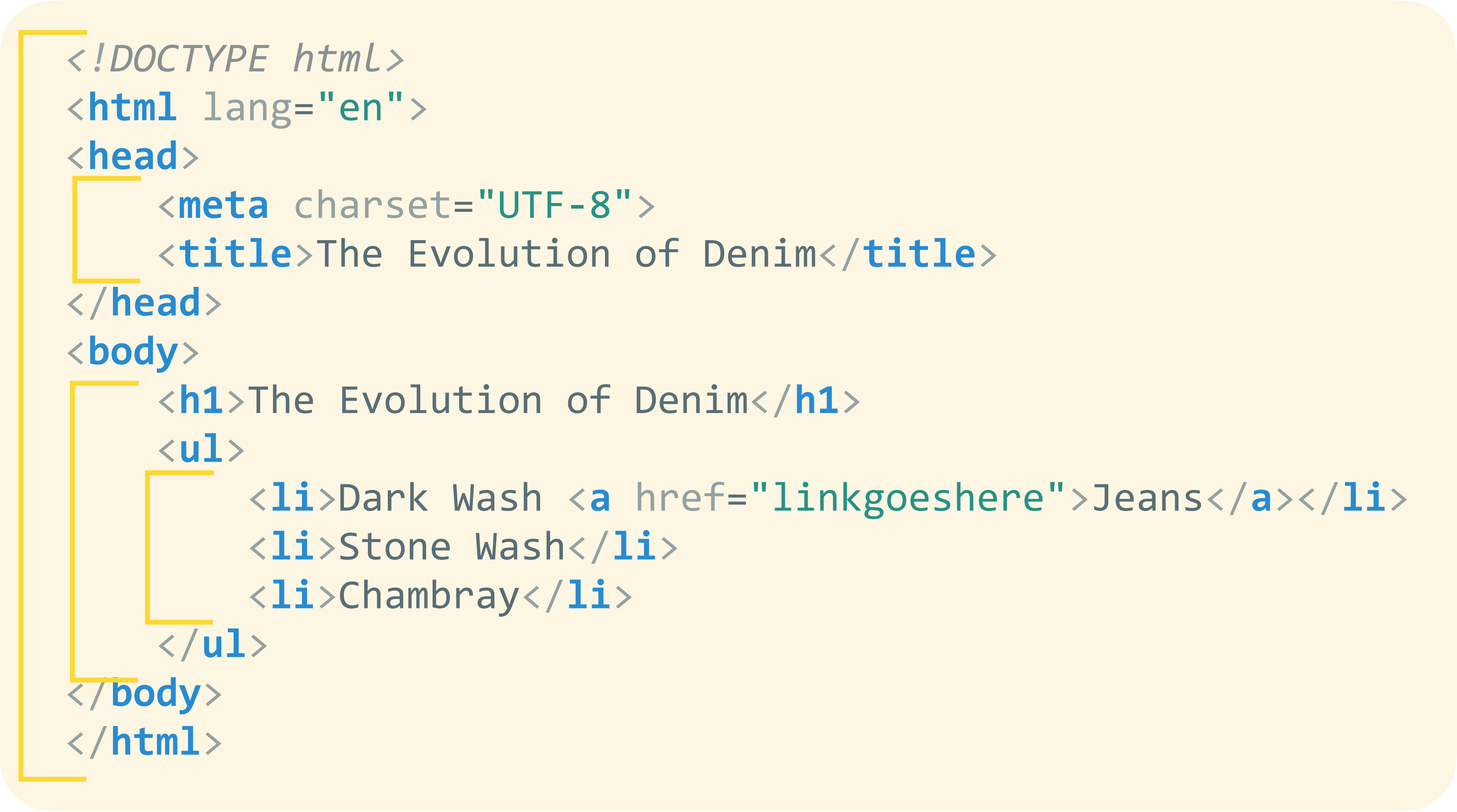
TIMING

3 min

1. Look at the starter code in the “dom_tree” folder
2. Pair up with a partner and diagram out the DOM tree
3. Keep in mind: 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.



```
<!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>
```

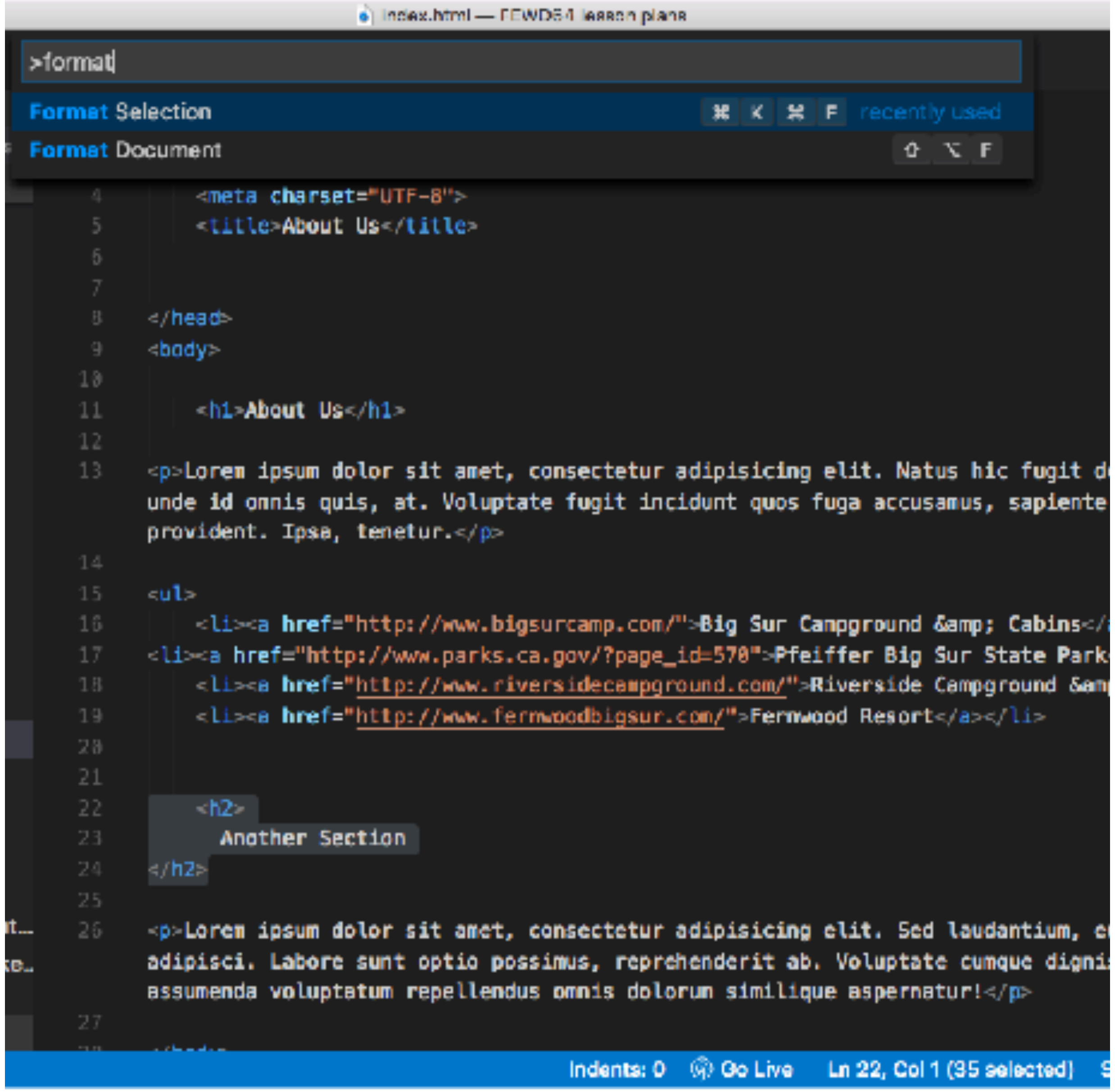
PAGE STRUCTURE — VISUAL STUDIO CODE HELPER!

Using the menu

- Press `command-shift-P`
- Type **format**
- Select from options
  - `Format Selection`
  - `Format Document`

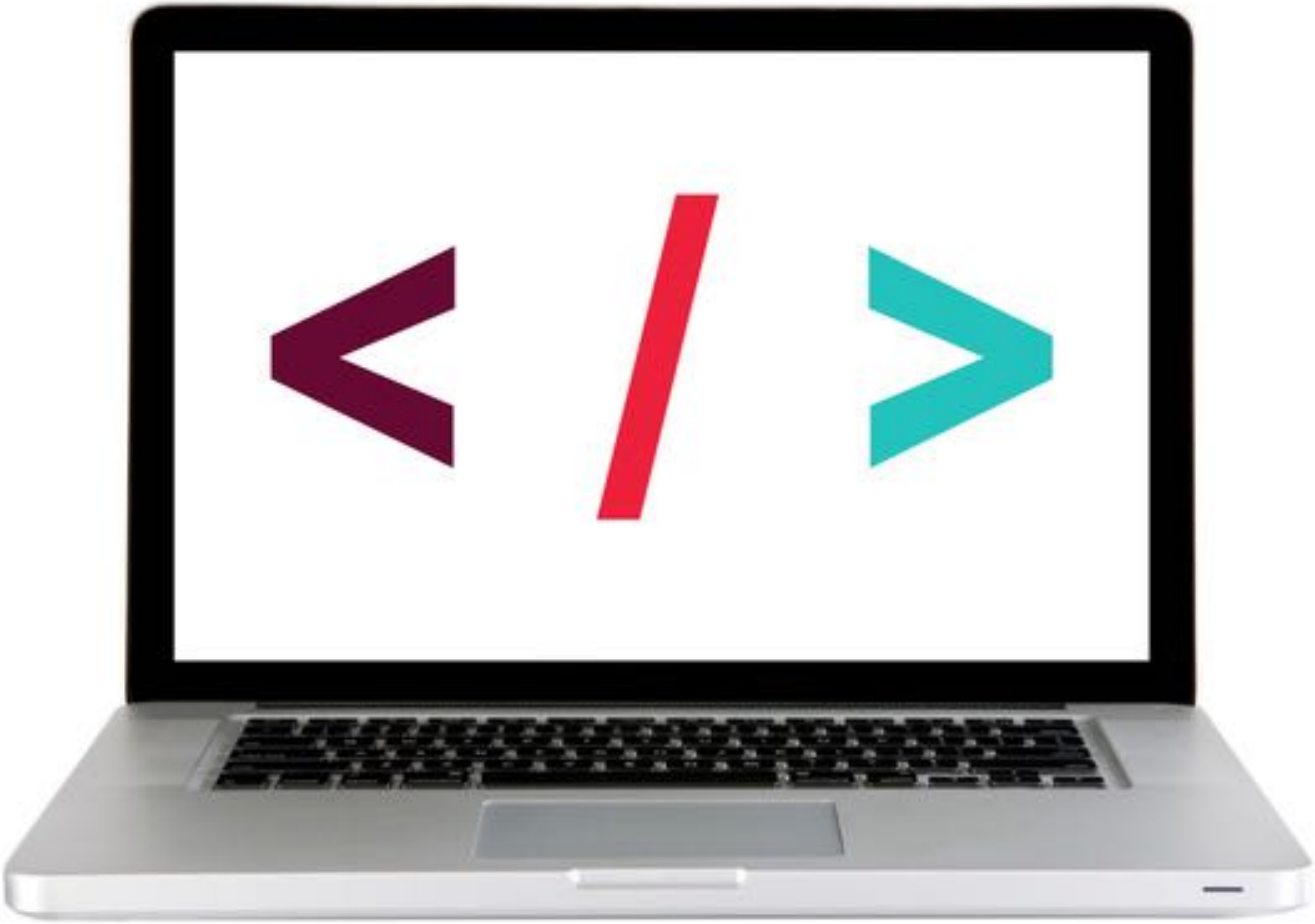
Keyboard shortcuts to format entire document:

- `shift-option-F` (Mac)
- `shift-alt-F` (Windows)


index.html — FEWD54 lesson plans
>format
Format Selection ⌘ K ⌘ F recently used
Format Document ⇧ ⌥ F
Indents: 0
Go Live
Ln 22, Col 1 (35 selected)

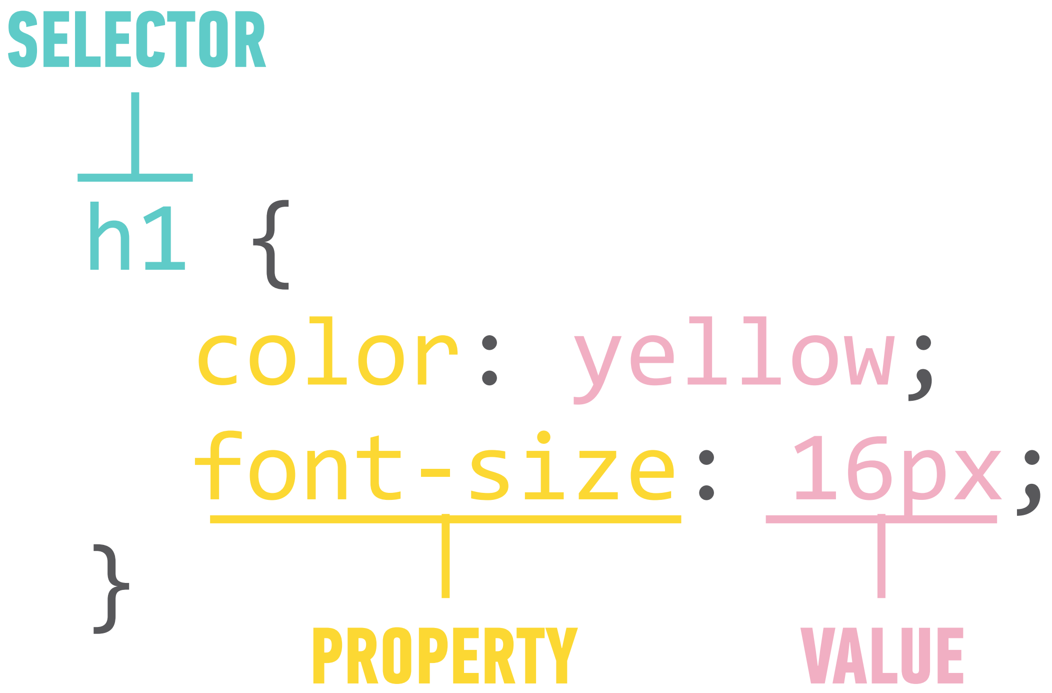
```
        <meta charset="UTF-8">
        <title>About Us</title>


</head>
<body>

        <h1>About Us</h1>

<p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Natus hic fugit d
unde id omnis quis, at. Voluptate fugit incidunt quos fuga accusamus, sapiente
provident. Ipsa, tenetur.</p>

<ul>
        <li><a href="http://www.bigsurcamp.com/">Big Sur Campground &amp; Cabins</
<li><a href="http://www.parks.ca.gov/?page_id=570">Pfeiffer Big Sur State Park
        <li><a href="http://www.riversidecampground.com/">Riverside Campground &am
        <li><a href="http://www.fernwoodbigsur.com/">Fernwood Resort</a></li>


        <h2>
          Another Section
</h2>

<p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Sed laudantium, e
adipisci. Labore sunt optio possimus, reprehenderit ab. Voluptate cumque digni
assumenda voluptatum repellendus omnis dolorum similique aspernatur!</p>
```

DEMO: UGLY INDENTING & VISUAL STUDIO CODE HELP!



MORE CSS SELECTORS

WHAT IS A SELECTOR?



SELECTOR

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

PROPERTY

VALUE

SELECTORS — MOST COMMON

- We've been using type selectors… but there are other kinds of selectors too:

SELECTOR:	MEANING:	EXAMPLE:
TYPE	Selects an element	`a {}`
DESCENDANT	Selects an element that is a **descendent** of another element	`p a {}`
UNIVERSAL	Selects **all** elements in a document	`* {}`
MULTIPLE	Selects multiple elements	`h1, h2 {}`

DESCENDANT SELECTORS

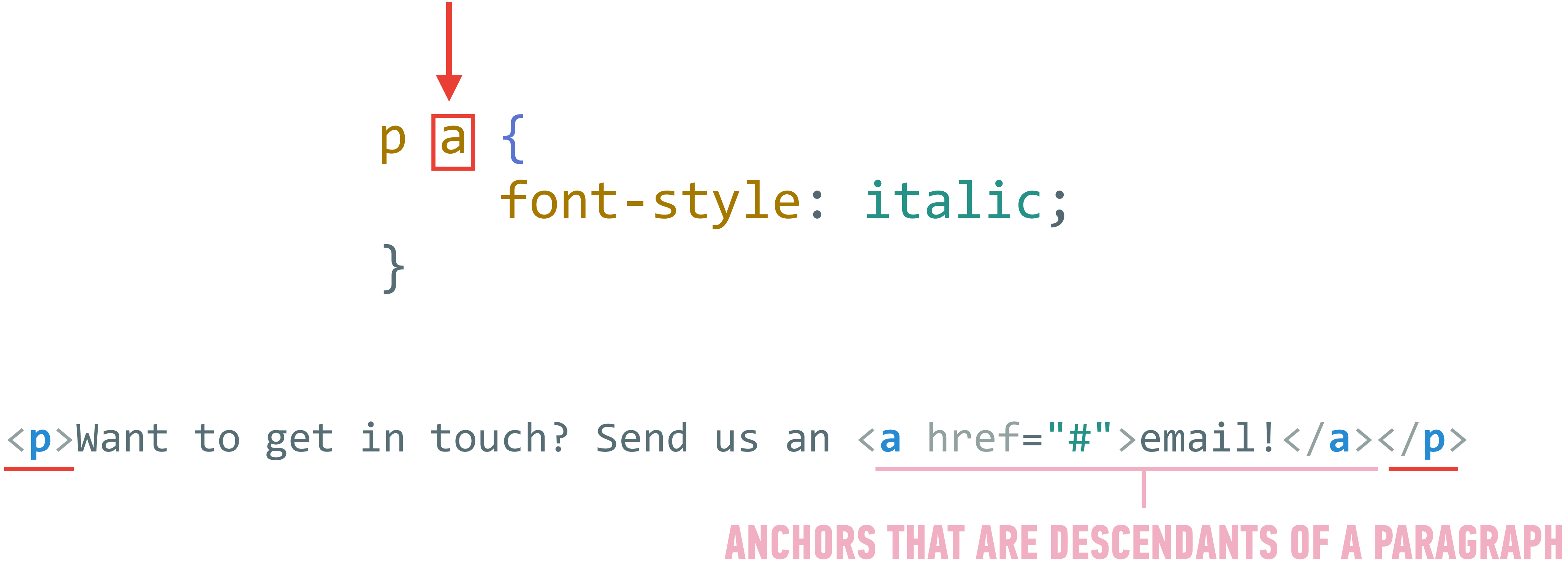
SELECTORS

Which element here will be italic?

```
p a {
    font-style: italic;
}
```

SELECTORS

- The **last** element in the selector string is *always* the one that we are styling.
- Try reading right-to-left!



```
p a {
    font-style: italic;
}
```

```
<p>Want to get in touch? Send us an <a href="#">email!</a></p>
```

ANCHORS THAT ARE DESCENDANTS OF A PARAGRAPH

SELECTORS

```html
<header>
    <h1><a href="">Nested Selectors</a></h1>
    <nav>
        <a href="">Home</a>
        <a href="">About</a>
        <a href="">Contact</a>
        <a href="">Blog</a>
    </nav>
</header>
```

ANCHORS THAT ARE DESCENDANTS OF A NAV.

NOTE: THEY'RE ALSO DESCENDANTS OF A HEADER TOO!

SELECTORS

ANCHORS THAT ARE DESCENDANTS OF A H1

```
<header>
    <h1><a href="">Nested Selectors</a></h1>
    <nav>
        <a href="">Home</a>
        <a href="">About</a>
        <a href="">Contact</a>
        <a href="">Blog</a>
    </nav>
</header>
```

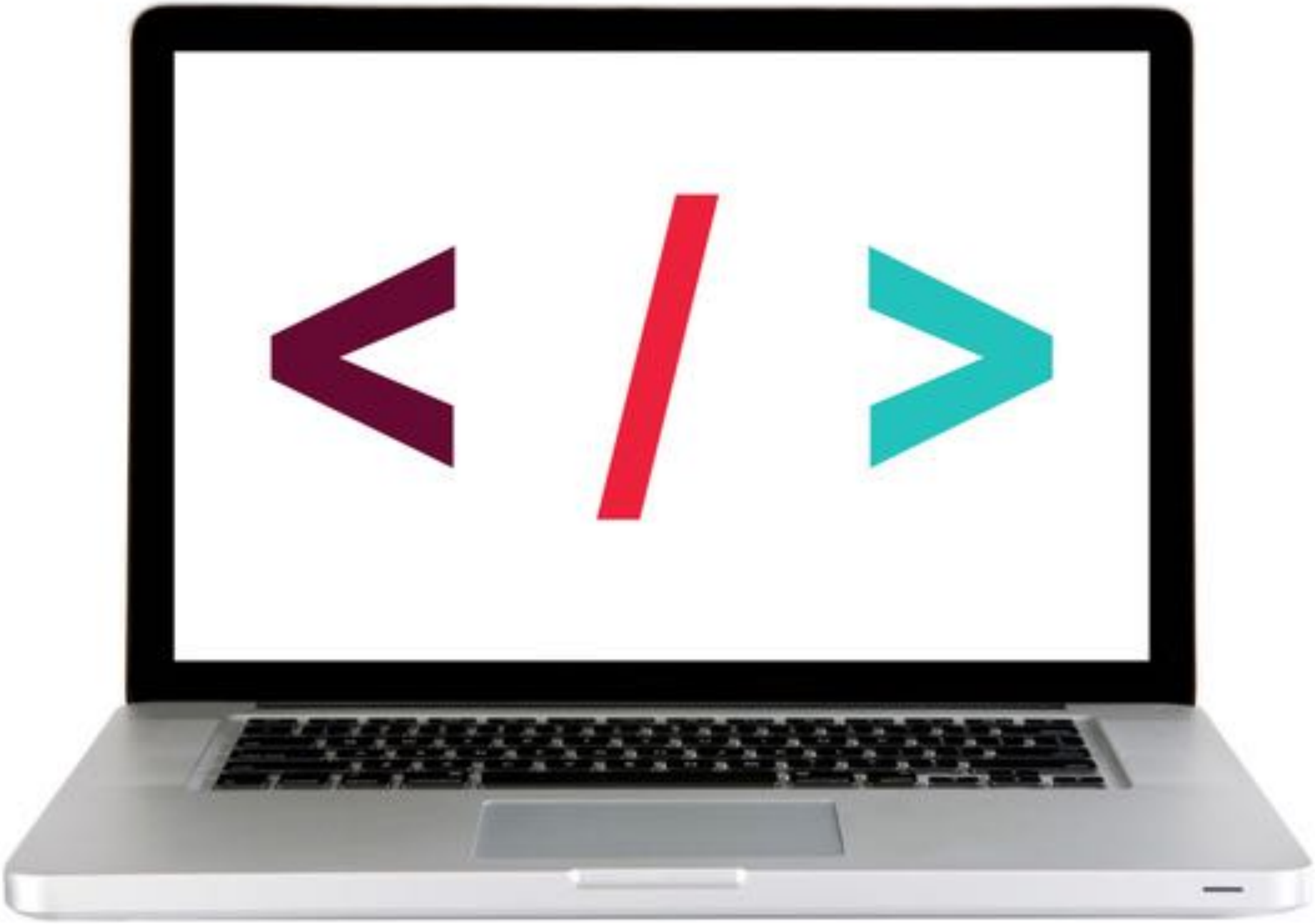
NOTE: THEY'RE ALSO DESCENDANTS OF A HEADER TOO!

SELECTORS

```
<footer>
    <nav>
        <a href="">About</a>
        <a href="">Contact</a>
        <a href="">Blog</a>
    </nav>
</footer>
```

WHAT ARE THESE ANCHORS DESCENDANTS OF?

CODEALONG: CASCADING EXAMPLE PART 1



ACTIVITY: DESCENDANT SELECTORS

EXERCISE

LOCATION

- starter_code folder > descendant_selectors folder

KEY OBJECTIVE

- Practice using descendant selectors

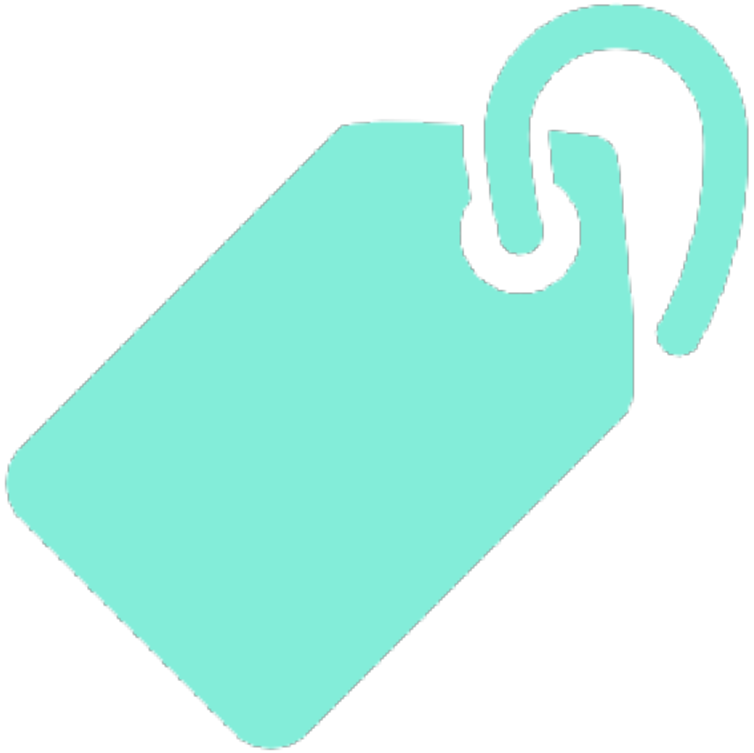
TIMING

5 min

1. Follow the instructions in the style.css file.
2. It should look like mocks.png if you did it correctly

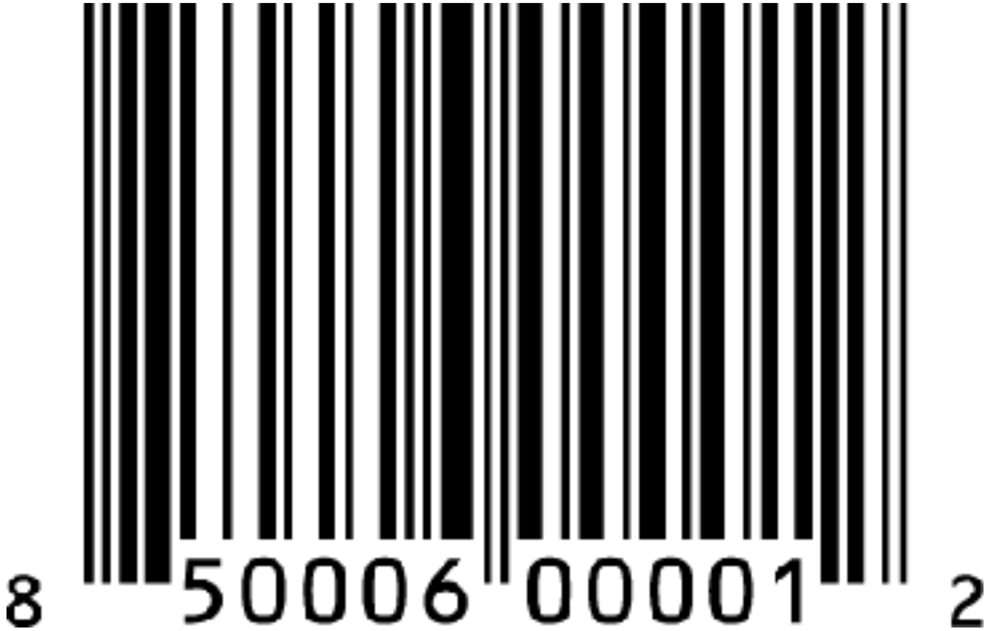
TARGETING SPECIFIC ELEMENTS: CLASSES AND IDS

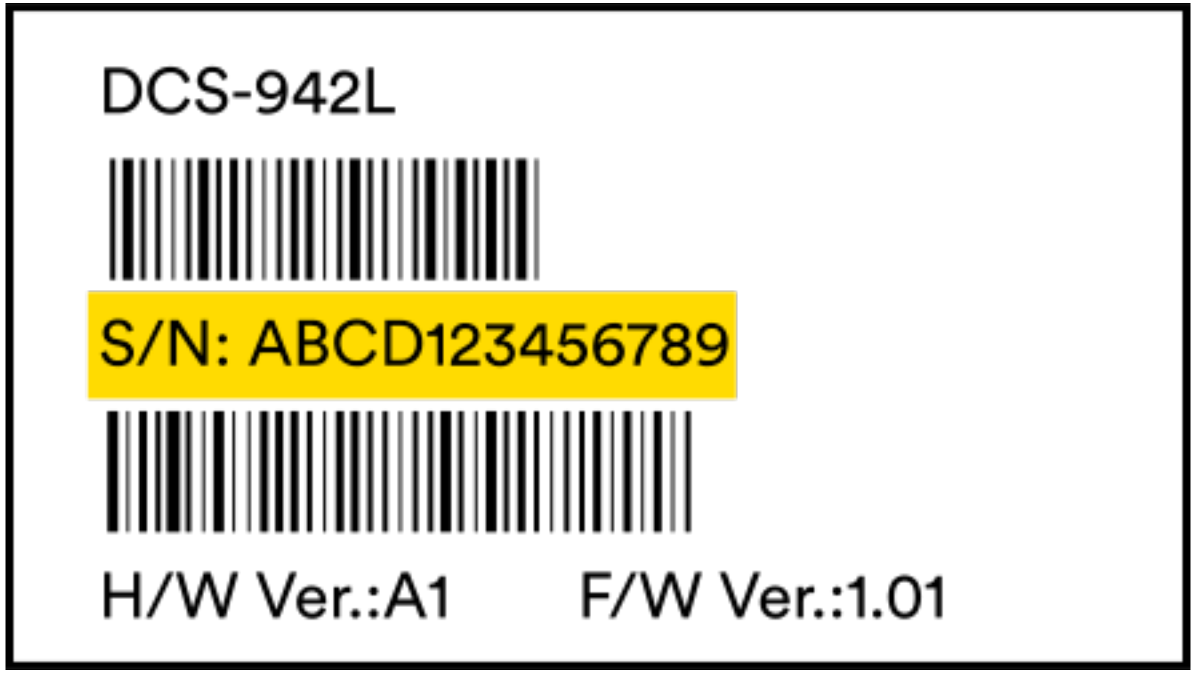
TARGETING SPECIFIC ELEMENTS



- Classes & IDs allow us to add ‘labels’ to elements so we can target them in our CSS.

WHY DO WE USE BAR CODES OR SERIAL NUMBERS?




DCS-942L

S/N: ABCD123456789

H/W Ver.:A1 F/W Ver.:1.01

CLASSES

- Classes are used to group elements together
- Like bar codes or UPCs (universal product codes)

```
<div class="alert">Content</div>
```

```
.alert {
    color: red;
    font-size: 20px;
}
```

CLASSES AND IDS

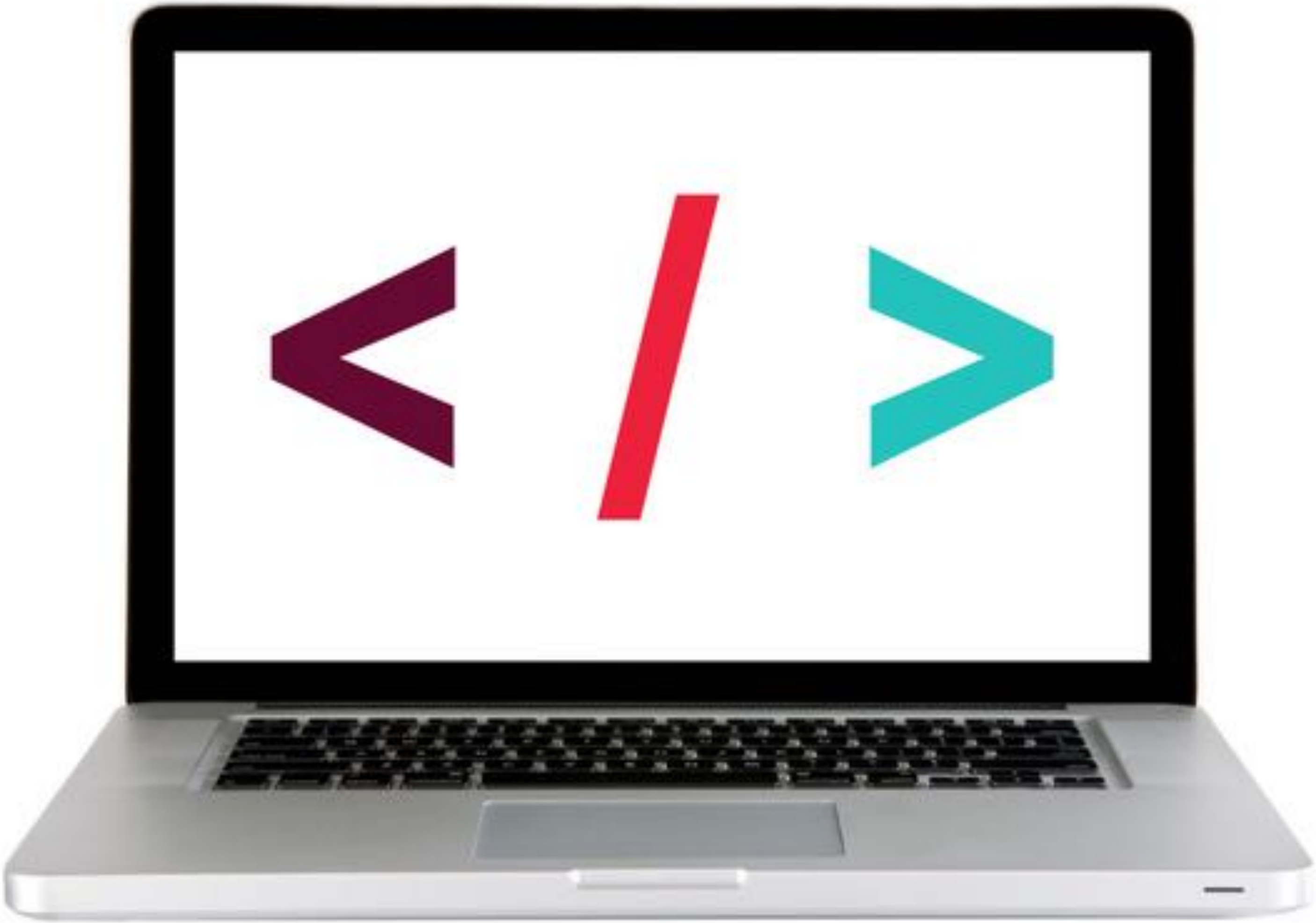
IDS

- Ids are used to target *one specific element*
- Each element can only have one id
- **Important**: two elements on the same page cannot have the same id

```html
<nav id="main-nav">Content</nav>
```

```css
#main-nav {
   text-align: center;
}
```

DEMO: CLASSES AND IDS EXAMPLE



ACTIVITY: CLASSES AND IDS ICE CREAM

EXERCISE

LOCATION

- `starter_code > classes_and_ids`

KEY OBJECTIVE

- Use classes and IDs to target elements

TIMING

1. Look at the image provided. Which ice cream items would you use a class to style? How about an ID?
2. Follow steps 1 - 3 in your CSS file (towards the end)

WHY IS CSS “CASCADING”?

INHERITANCE

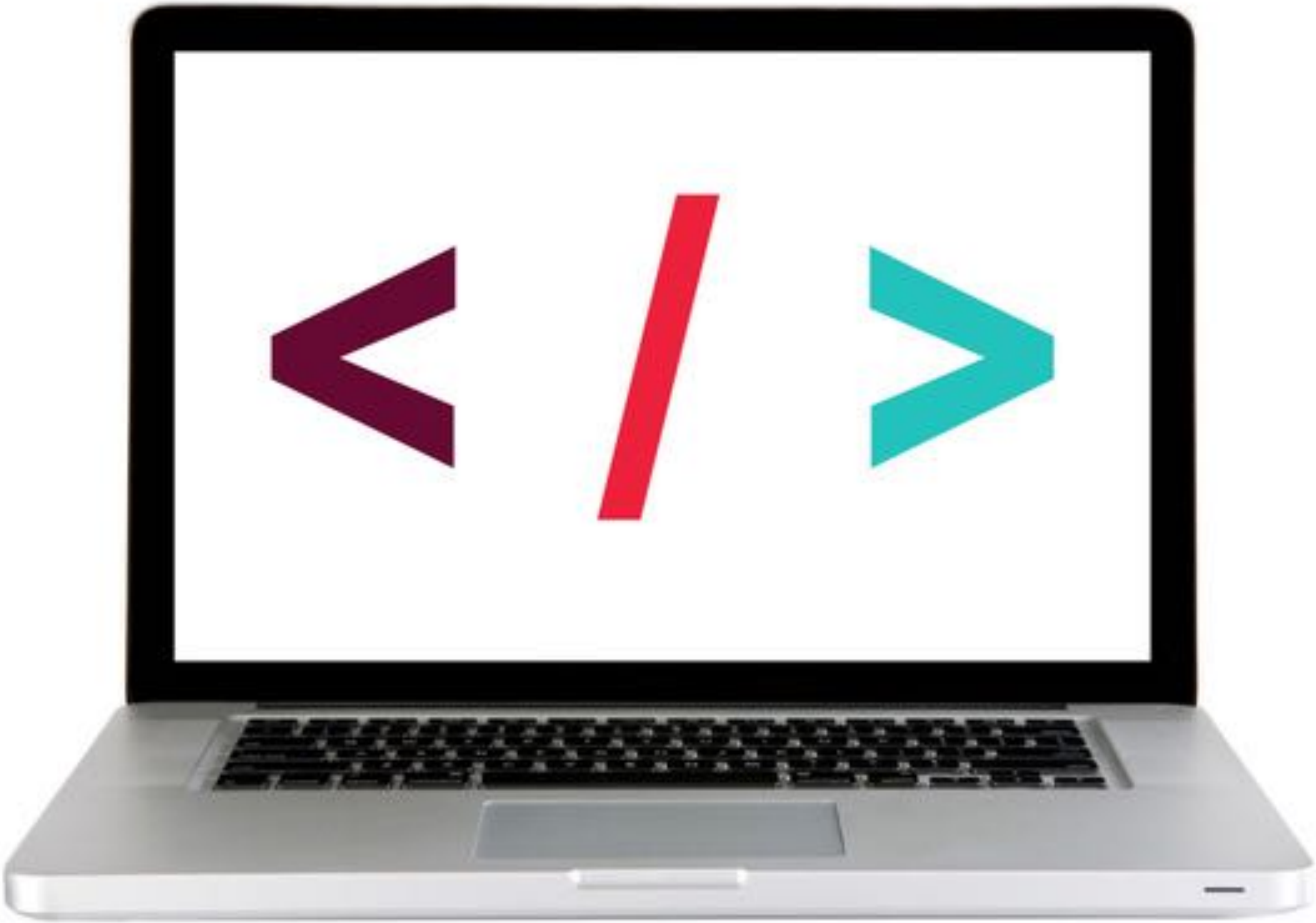
INHERITANCE — SETTING BASE STYLES

- 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 be inherited, or applied, to most child elements **unless there is a more specific rule that applies**.

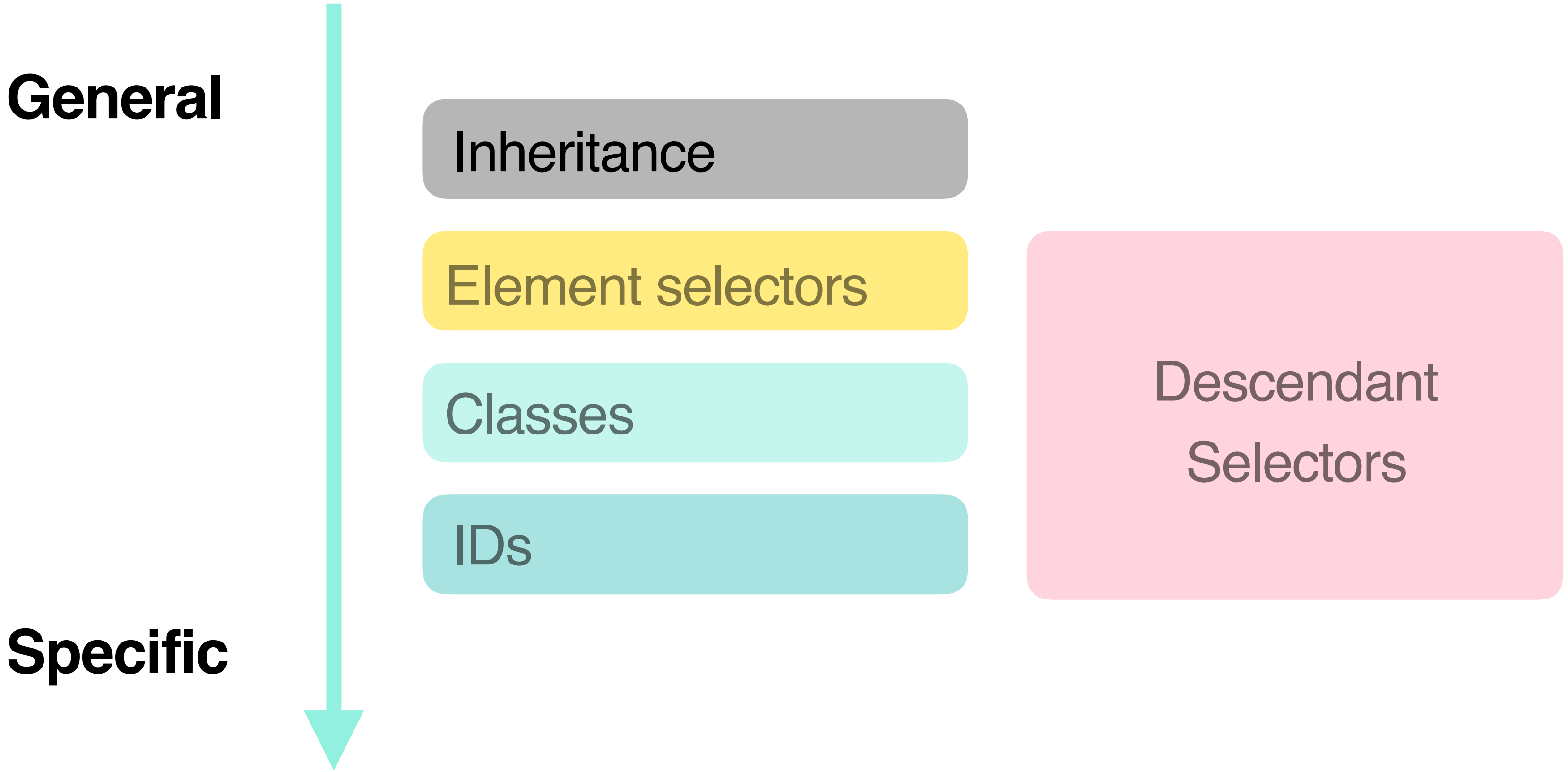
Inherited properties you'll use in this course:

- `color`
- `font-family`
- `font-size`
- `font-style`
- `font-weight`
- `letter-spacing`
- `line-height`
- `list-style`
- `text-align`
- `text-indent`
- `text-transform`
- `word-spacing`

CODEALONG: CASCADING EXAMPLE PART 2



MORE ABOUT CASCADING — GENERAL TO MORE SPECIFIC



ACTIVITY: DISCUSS INHERITANCE

EXERCISE

KEY OBJECTIVE

- Explain inheritance in CSS. How can you use inheritance in your stylesheets to write less code in the long run?

TYPE OF ACTIVITY

- Turn and Talk

TASKS

1 min 1. Discuss the question with your groups

2 min 2. Pick one person to jot down your thoughts and share them via Slack.

ACTIVITY: SPECIFICITY CACTUS SITE

EXERCISE

KEY OBJECTIVE

- Practice using inheritance to set up default styles for a webpage

LOCATION

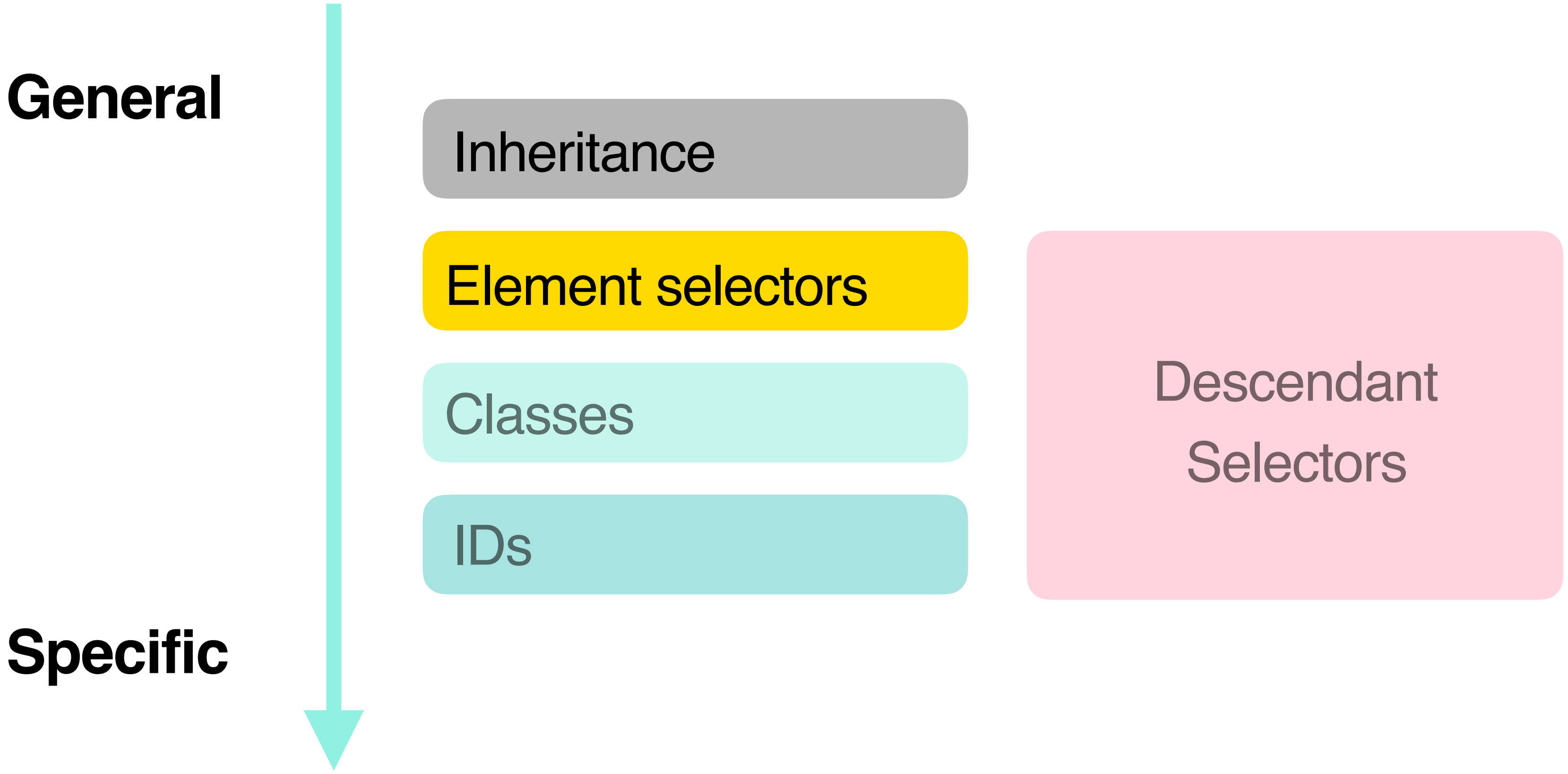
- Starter Code > Specificity

TASKS

1 min	1. Follow instructions under STEP 1

ELEMENT SELECTORS

MORE ABOUT CASCADING — GENERAL TO MORE SPECIFIC



General

Inheritance

Element selectors

Classes

IDs

Specific

Descendant Selectors

ACTIVITY: SPECIFICITY CACTUS SITE

EXERCISE

KEY OBJECTIVE

- Practice using element selectors to style groups of elements

TYPE OF ACTIVITY

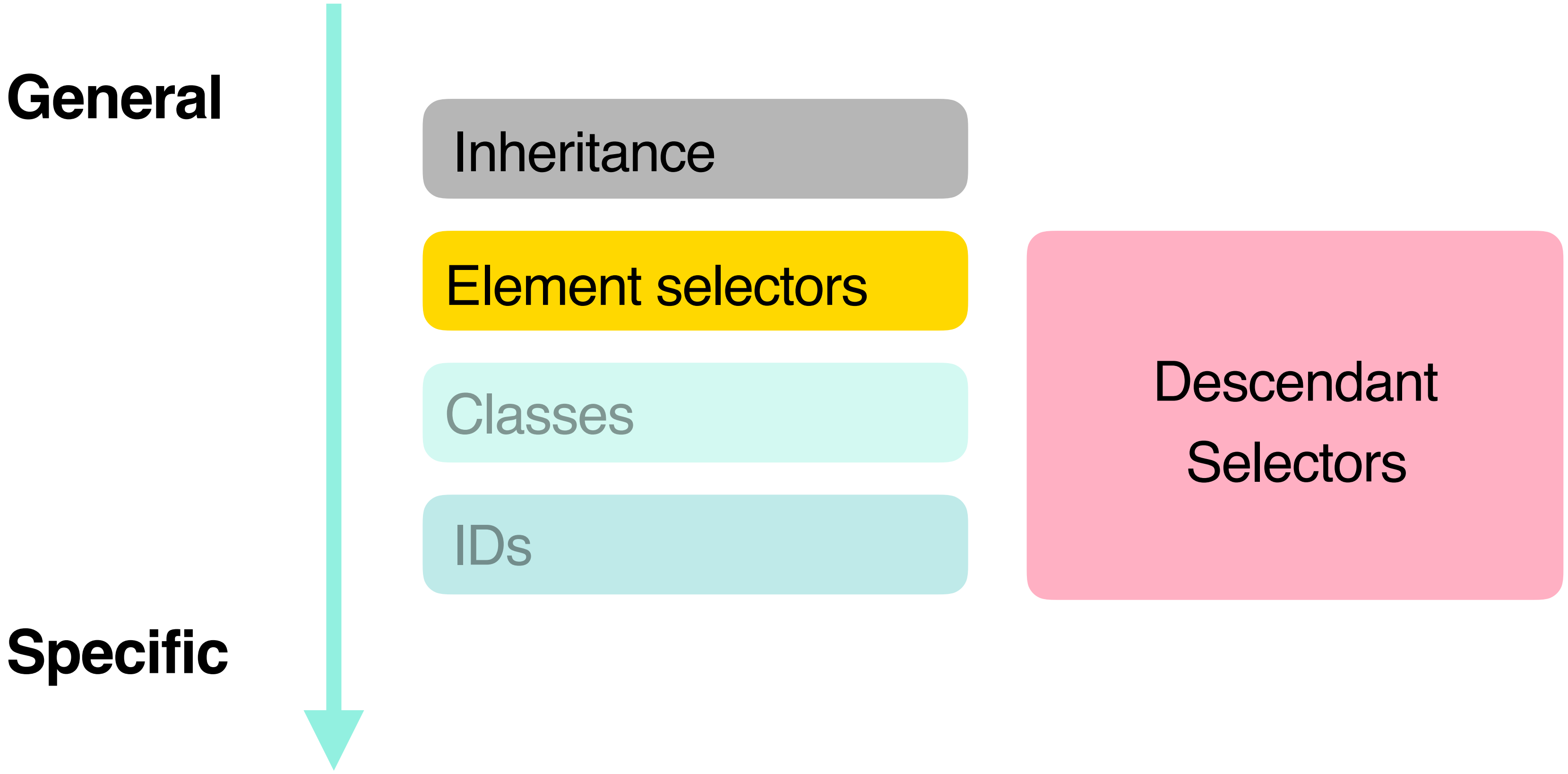
- Starter Code > Specificity

TASKS

1 min 1. Follow the instructions under STEP 2.

DESCENDANT SELECTORS

MORE ABOUT CASCADING — GENERAL TO MORE SPECIFIC



ACTIVITY: SPECIFICITY CACTUS SITE

EXERCISE

KEY OBJECTIVE

- Practice using nested selectors to style elements based on their relationships.

TYPE OF ACTIVITY

- Starter Code > Specificity

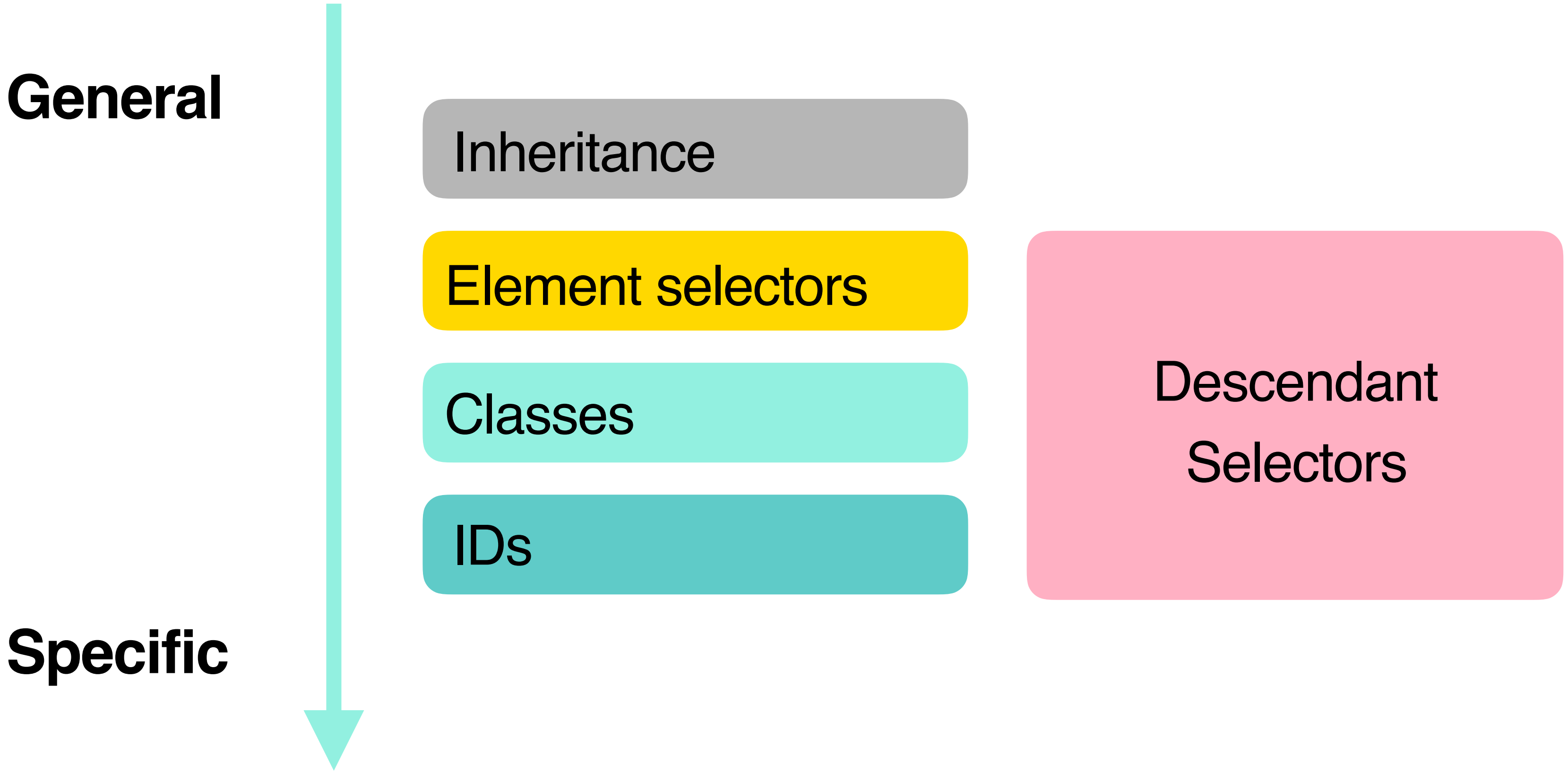
TASKS

1 min

1. Follow the instructions under STEP 3.

CLASSES AND IDS

MORE ABOUT CASCADING — GENERAL TO MORE SPECIFIC



ACTIVITY: SPECIFICITY CACTUS SITE

EXERCISE

KEY OBJECTIVE

- Practice using classes and IDs to style unique elements or groups of elements

TYPE OF ACTIVITY

- Starter Code > Specificity

TASKS

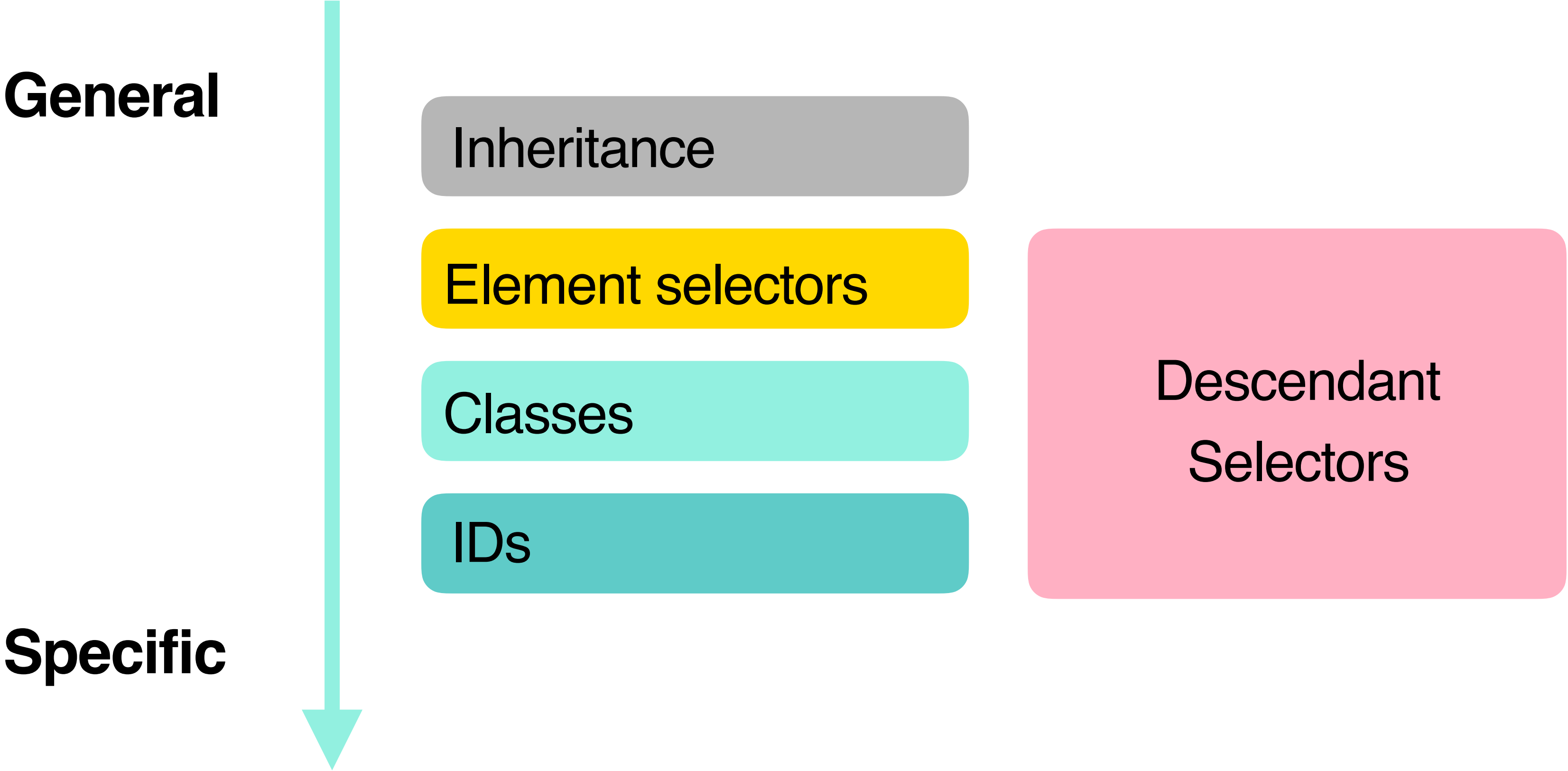
1 min 1. Follow the instructions under STEP 4.

SPECIFICITY GAME!!!

MORE ABOUT CASCADING — GENERAL TO MORE SPECIFIC

SPECIFICITY:

The *more specific rule* will take precedence over the more general rule



General

Inheritance

Element selectors

Classes

IDs

Descendant Selectors

Specific

CSS CASCADING — THE SPECIFICITY GAME

- If multiple style rules are targeted at the same element, which style will be applied?
- We can calculate the specificity of the selectors to find out!
- The styles for the more specific selector are the styles that will be applied.

```
p a {
    font-size: 50px;
}

a {
    font-size: 30px;
}
```

CSS CASCADING — THE SPECIFICITY GAME

TO CALCULATE SPECIFICITY,
TAKE A LOOK AT THE ENTIRE "SELECTOR CHAIN"

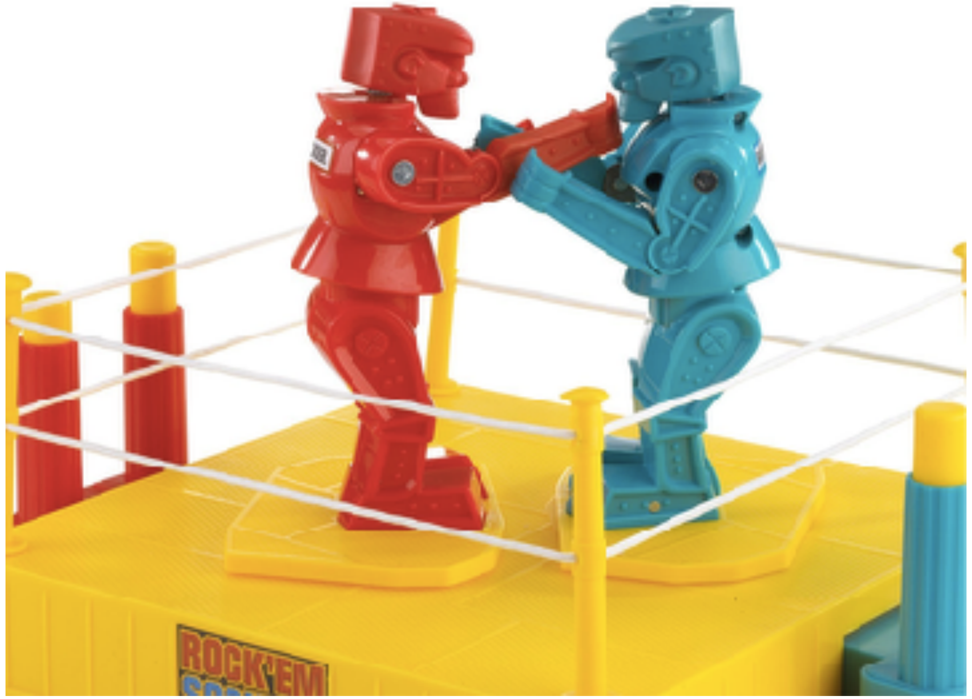
```
p a {
    font-size: 50px;
}
```

CSS CASCADING — THE SPECIFICITY GAME

If you have conflicting styles under the “p a” selector and the “a” selector… who will win?

p a

a



CSS CASCADING — THE SPECIFICITY GAME

Use a chart like this to calculate the specificity to find out!

IDs	Classes	Elements

CSS CASCADING — THE SPECIFICITY GAME

2 ELEMENTS

```
p a {
    font-size: 50px;
}
```

IDs	Classes	Elements
0	0	2

SCORE: 2

CSS CASCADING — THE SPECIFICITY GAME

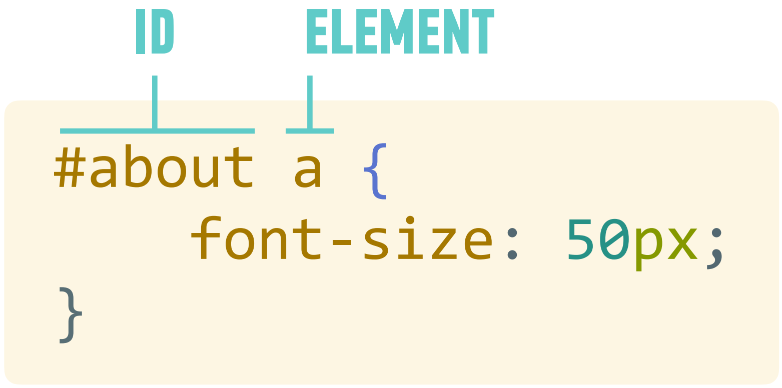
1 CLASS

```
.about {
    font-size: 50px;
}
```

IDs	Classes	Elements
0	1	0

SCORE: 10

CSS CASCADING — THE SPECIFICITY GAME



```
#about a {
    font-size: 50px;
}
```

IDs	Classes	Elements
1	0	1

SCORE: 101

CSS CASCADING — THE SPECIFICITY GAME

Write this down!

IDs	Classes	Elements

CSS CASCADING — THE SPECIFICITY GAME

SLACK POLL: *Will the font-size for the anchor be 50px or 30px?*

HTML:

```html
<p>Visit my <a href="#">Website</a></p>
```

CSS:

```css
p a {
    font-size: 50px;
}

a {
    font-size: 30px;
}
```

CSS CASCADING — THE SPECIFICITY GAME

p a

IDs	Classes	Elements
0	0	2

a

IDs	Classes	Elements
0	0	1

WINNER!

CSS CASCADING — THE SPECIFICITY GAME

SLACK POLL: *Will the font-size for the anchor be 50px or 30px?*

HTML:

```html
<p>Visit my <a href="#">Website</a></p>
```

CSS:

```css
p a { /* Score: 2 */
  font-size: 50px;
}

a { /* Score: 1 */
  font-size: 30px;
}
```

CSS CASCADING — THE SPECIFICITY GAME

SLACK POLL: *Will the anchor with the class* `home` *be pink or blue?*

HTML:

```html
<nav id="main-nav">
    <a href="#" class="home">Home</a>
    <a href="#">About</a>
    <a href="#">Resume</a>
</nav>
```

CSS:

```css
#main-nav a {
    color: pink;
}

.home {
    color: blue;
}
```

CSS CASCADING

`#main-nav a`

IDs	Classes	Elements
1	0	1

WINNER!

`.home`

IDs	Classes	Elements
0	1	0

CSS CASCADING — THE SPECIFICITY GAME

SLACK POLL: *Will the anchor with the class* home *be pink or blue?*

HTML:

```html
<nav id="main-nav">
    <a href="#" class="home">Home</a>
    <a href="#">About</a>
    <a href="#">Resume</a>
</nav>
```

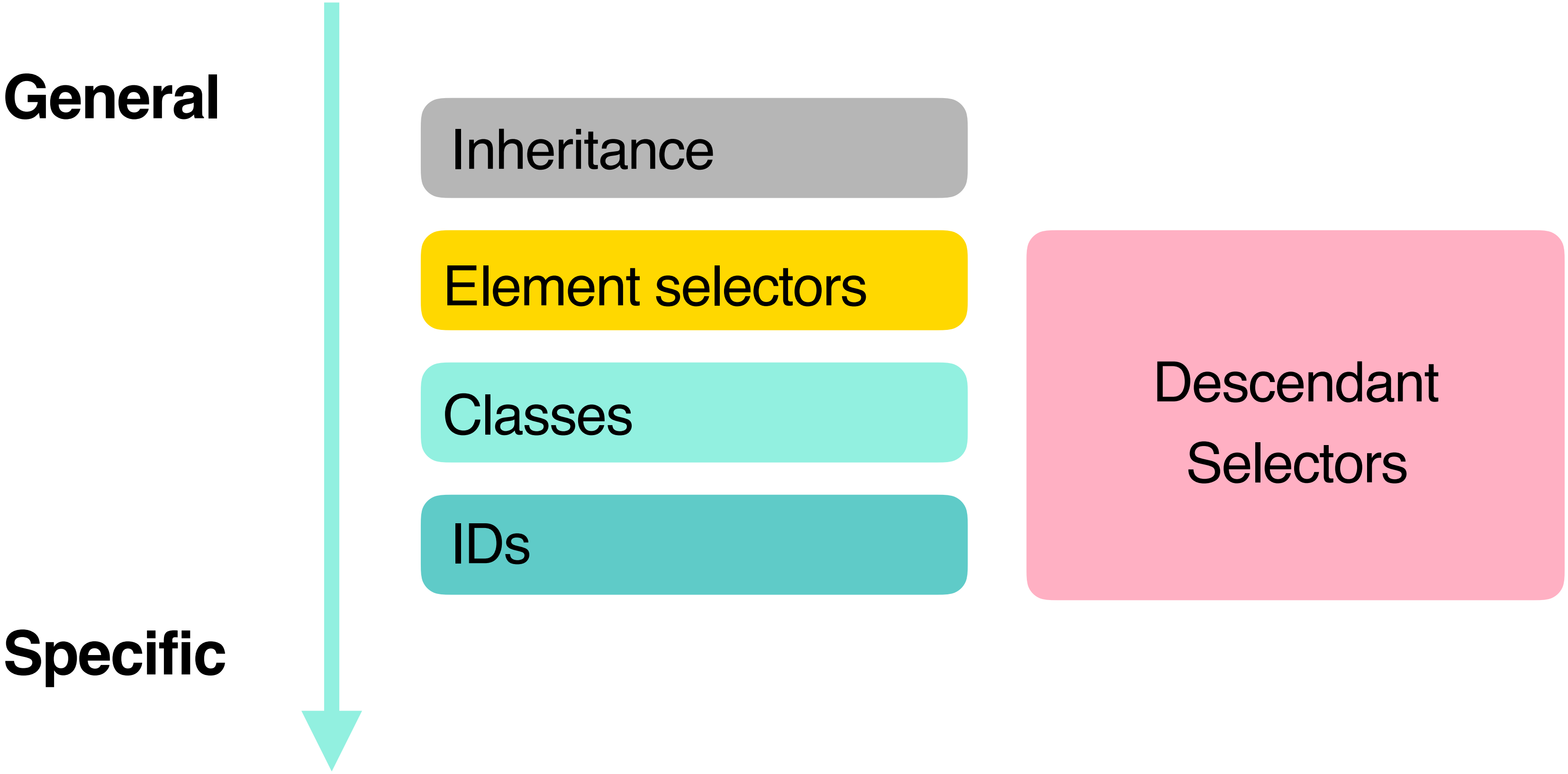
CSS:

```css
#main-nav a { /* Score: 101 */
    color: pink;
}

.home { /* Score: 10 */
    color: blue;
}
```

MORE ABOUT CASCADING — GENERAL TO MORE SPECIFIC

SPECIFICITY:

The *more specific rule* will take precedence over the more general rule



LAST RULE

CSS CASCADING — THE SPECIFICITY GAME

SLACK POLL: *Will the anchor have an underline or no underline?*

HTML:

```html
<li>
    <p>Visit this <a href="#">cool</a> site.</p>
</li>
```

CSS:

```css
li a { /* Score: 2 */
    text-decoration: underline;
}

p a { /* Score: 2 */
    text-decoration: none;
}
```

LAST RULE

`li a`

IDs	Classes	Elements
0	0	2

`p a`

IDs	Classes	Elements
0	0	2

TIE?!?

SLACK POLL: *Will the anchor have an underline or no underline?*

HTML:

```html
<li>
    <p>Visit this <a href="#">cool</a> site.</p>
</li>
```

CSS:

```css
li a { /* Score: 2 */
    text-decoration: underline;
}

p a { /* Score: 2 */
    text-decoration: none;
}
```

TIEBREAKER? LAST RULE WINS!

CSS CASCADING — THE SPECIFICITY GAME

Let’s say we switched it…

HTML:

```html
<li>
    <p>Visit this <a href="#">cool</a> site.</p>
</li>
```

CSS:

```css
p a { /* Score: 2 */
    text-decoration: none;
}

li a { /* Score: 2 */
    text-decoration: underline;
}
```

TIEBREAKER? LAST RULE WINS!

ACTIVITY

EXERCISE

KEY OBJECTIVE

- When multiple style rules apply to the same element, how can we figure out which one will be applied?
- How can we apply this principle to allow us to write less CSS in the long run?

TYPE OF ACTIVITY

- Turn and Talk

TASKS

5 min

1. Talk with your groups. Have one person write out your responses in Slack.

THINGS YOU SHOULD USE IF YOU WANT TO BE

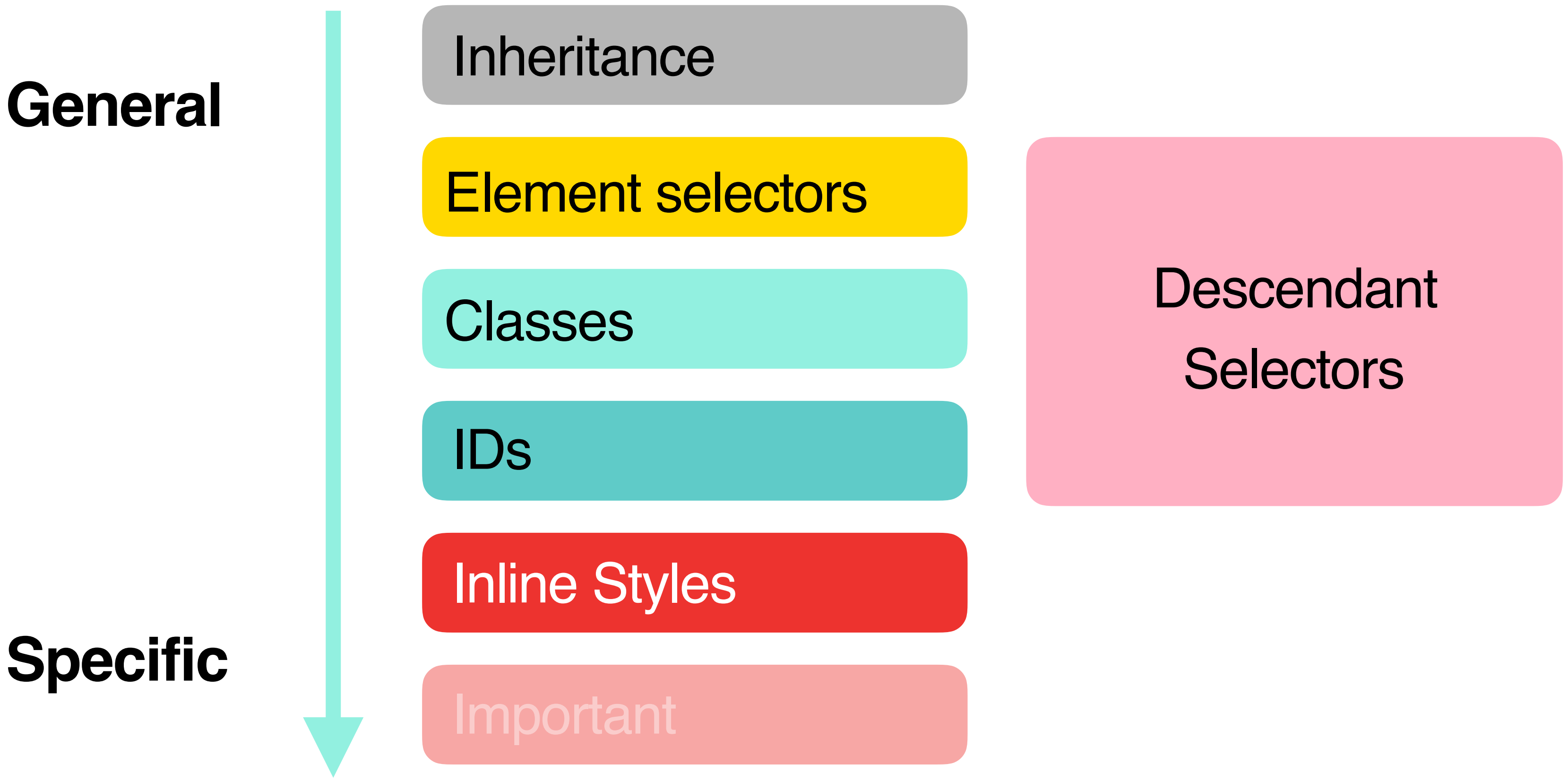
😭

INLINE STYLES

```html
<li style="color: red;">Content</li>
```

BAD!!!!

MORE ABOUT CASCADING



General

Specific

Inheritance

Element selectors

Classes

IDs

Inline Styles

Important

Descendant Selectors

CSS IMPORTANCE

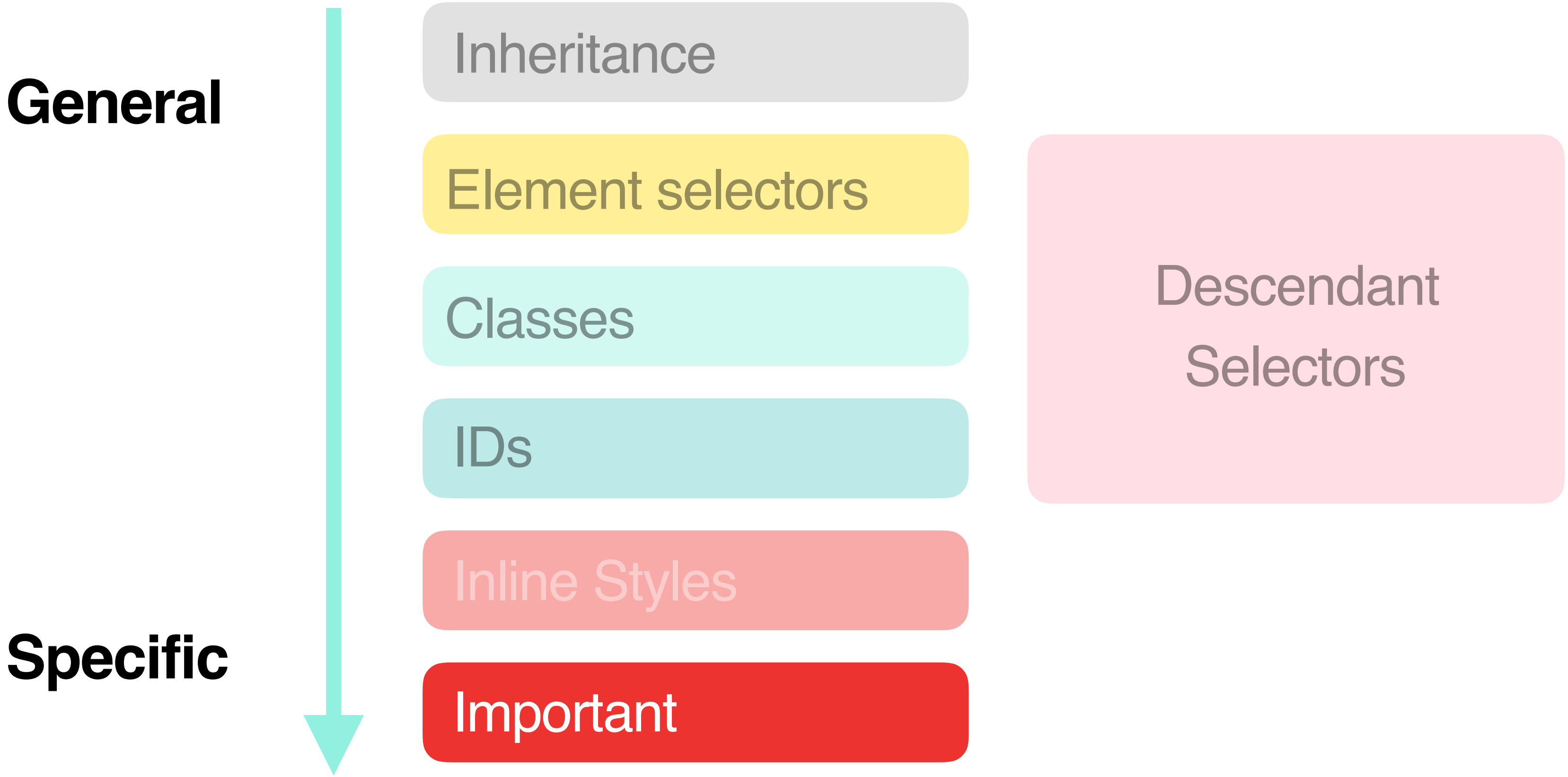
Adding **!important** after any property value indicates that it should be considered *more important than other rules that apply to the same element.*

```css
li {
   font-size: 100px !important;
}
```

ONLY WHEN NECESSARY!!!

(Which is almost never)

MORE ABOUT CASCADING



General

Specific

Inheritance

Element selectors

Classes

IDs

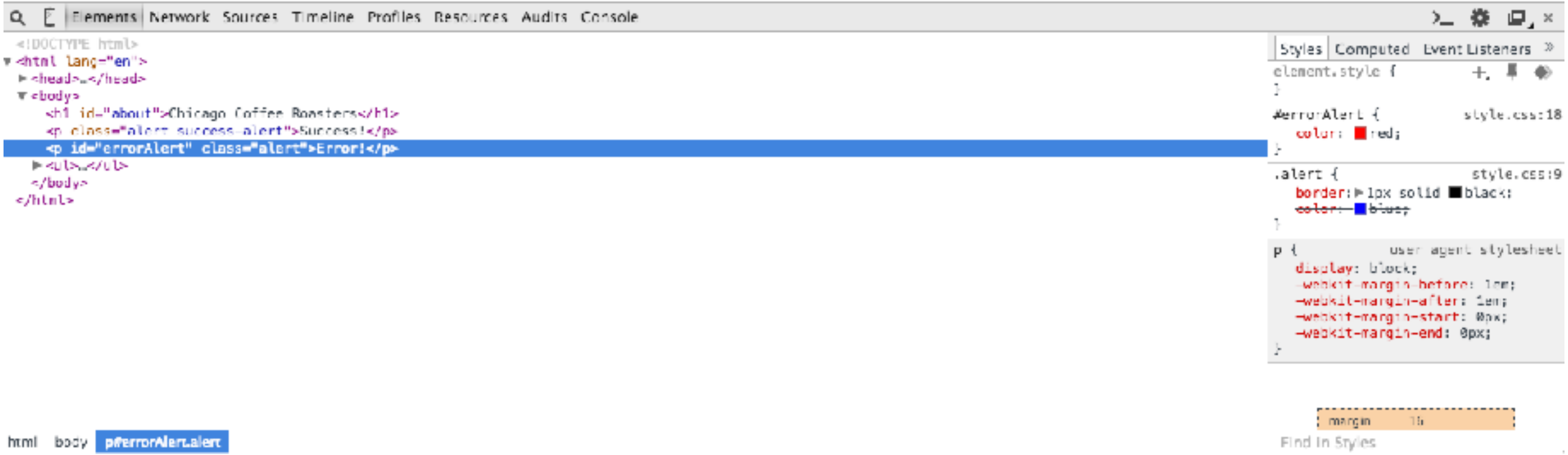
Inline Styles

Important

Descendant Selectors

TOOLS: BROWSER DEVELOPER TOOLS

CHROME DEV TOOLS



```
<!DOCTYPE html>
▼<html lang="en">
  ▶<head>…</head>
  ▼<body>
      <h1 id="about">Chicago Coffee Roasters</h1>
      <p class="alert success-alert">Success!</p>
      <p id="errorAlert" class="alert">Error!</p>
    ▶<ul>…</ul>
    </body>
</html>
```

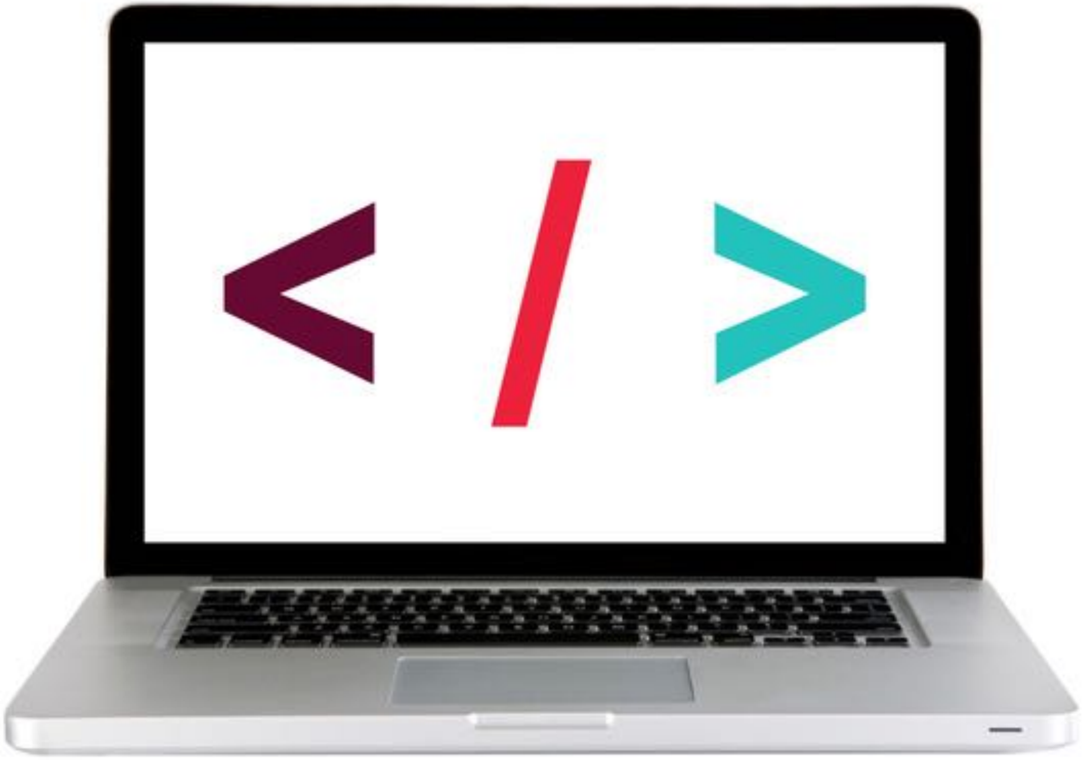
```
element.style {
}

#errorAlert {                style.css:18
   color: 🟥 red;
}

.alert {                      style.css:9
   border: ▶ 1px solid ■ black;
   ~~color: 🟦 blue;~~
}

p {                  user agent stylesheet
   display: block;
   -webkit-margin-before: 1em;
   -webkit-margin-after: 1em;
   -webkit-margin-start: 0px;
   -webkit-margin-end: 0px;
}
```

html body p#errorAlert.alert

Right click > Inspect

- WARNING: You can fiddle around with the styles here, but MAKE SURE TO UPDATE (and save) YOUR CSS FILE IN YOUR EDITOR!

DEMO: WORKING WITH CHROME DEV TOOLS



Use Dev Tools to inspect solutions!

LAB: MONUMENT BLOG

OVERVIEW

Congratulations! You have been hired as the developer for Monument Lifestyle Magazine. You are working closely with a designer and have been provided design mockups for the website. Your goal is to replicate these mockups as closely as possible.

You will be adding components to the blog page in the labs/ homework this week and will be working on building the landing page for the site during the lab sessions next week.

ACTIVITY: MONUMENT BLOG PLANNING

EXERCISE

KEY OBJECTIVE

- Plan out the Monument Magazine site.

LOCATION

- Starter code > Monument Blog > mocks_basic.png

TASKS

Consider the following questions with your groups:

1. What is the most common font-family?
2. Is there a common color for the text?
3. Is most of the text uppercase? Lowercase?
4. How about the anchors? Are they underlined?
5. Is there a common color for the anchors?

ACTIVITY: MONUMENT BLOG

EXERCISE

KEY OBJECTIVE

- Recreate the Monument Magazine site

TASKS

1. Look through the provided HTML.
2. Start by adding styles that are the most common to the body, using mocks_body.png and mocks_basic.png as a guide
3. **Bonus 1:** Work from mocks_medium.png
4. **Bonus 2:** Work from mocks_advanced.png (You'll need to do some research to implement a background-image in the header and FontAwesome icons in the footer)
5. **Bonus 3:** Add a hover effect to links

LEARNING OBJECTIVES

- Describe inheritance in CSS.
- Add classes and IDs to HTML elements and apply CSS styles to elements based on class and ID.
- Explain when you would use a class and when you would use an ID.
- Apply CSS to elements based on their relationships.

WEEKLY OVERVIEW

WEEK 2	CSS Selectors / Layout
WEEK 3	Layout Lab / Grid Systems & Wireframing
WEEK 4	Responsive Design / CSS Positioning

EXIT TICKETS