LET'S GET EVERYTHING SET UP!

1. Open the FEWD 54 website (https://svodnik.github.io/fewd54), click the Lessons tab, click the WEEK 2 button, and download the Lesson 3 starter code and slides. You'll want to keep the website open for other links and resources we'll be referencing in class.
2. Move the starter code and slides from your Downloads folder to the **fewd** folder you created last week.
3. Double-click on starter code folder to unzip it
4. 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!
5. Open the entire **fewd** folder with your code editor (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.
6. In Slack, join the **#03-css-selectors** channel.

WEEKLY OVERVIEW

WEEK 1	HTML Basics / CSS Basics
WEEK 2	CSS Selectors / Layout
WEEK 3	Lab / Refactoring & Lab

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.

AGENDA

- Review
- Relative URLs — Part 2
- CSS Selectors
- Lab Time

TODAY'S LAB:

NEXT CLASS:

REVIEW

ACTIVITY

EXERCISE

KEY OBJECTIVE

- Review Homework, Show off Portfolio Sites

TYPE OF EXERCISE

- Groups of 3

TIMING

10 min

1. Open Portfolio site on laptops and display them proudly!
2. Give feedback to your peers: "I like" and "I wish/wonder"
3. Were you able to complete any of the bonus tasks? Webfont, hover effect, favicon? Show your group how you did it!
4. Pick someone to jot down any questions your group might have.

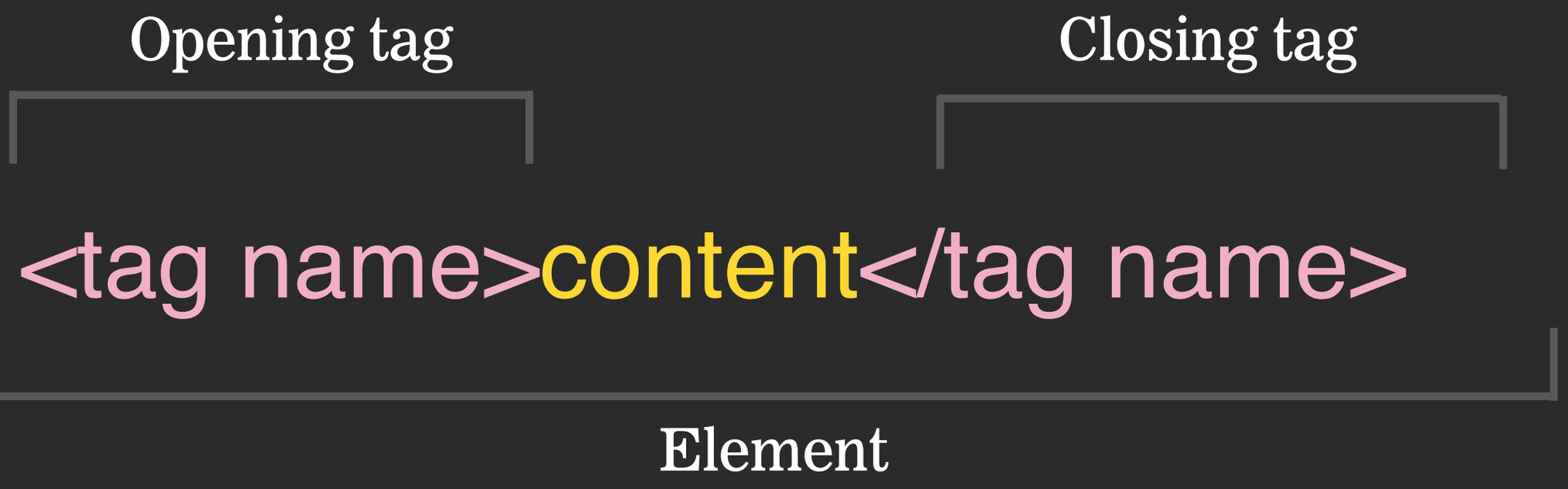
HOMEWORK BONUS

1. Add a webfont
2. Add a hover effect to anchors
3. Add a favicon

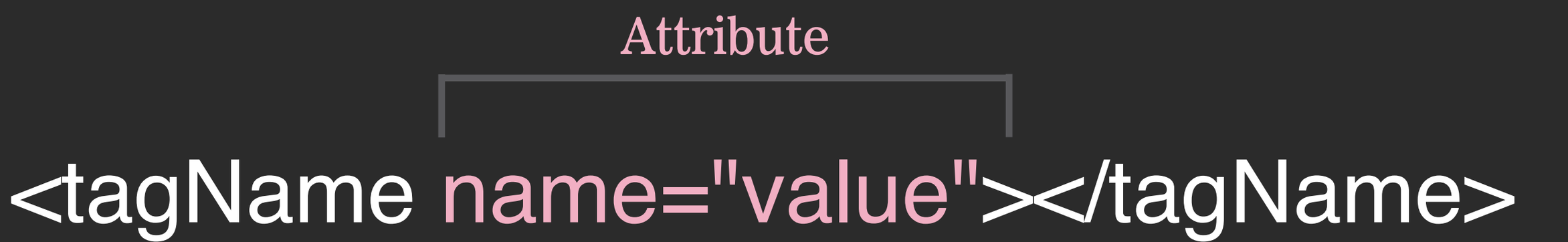
PSEUDO CLASSES

```
/* unvisited link */
a:link {
    color: #FF0000;
}

/* visited link */
a:visited {
    color: #00FF00;
}

/* mouse over link */
a:hover {
    color: #FF00FF;
}

/* selected link */
a:active {
    color: #0000FF;
}
```

HTML SYNTAX — TAGS



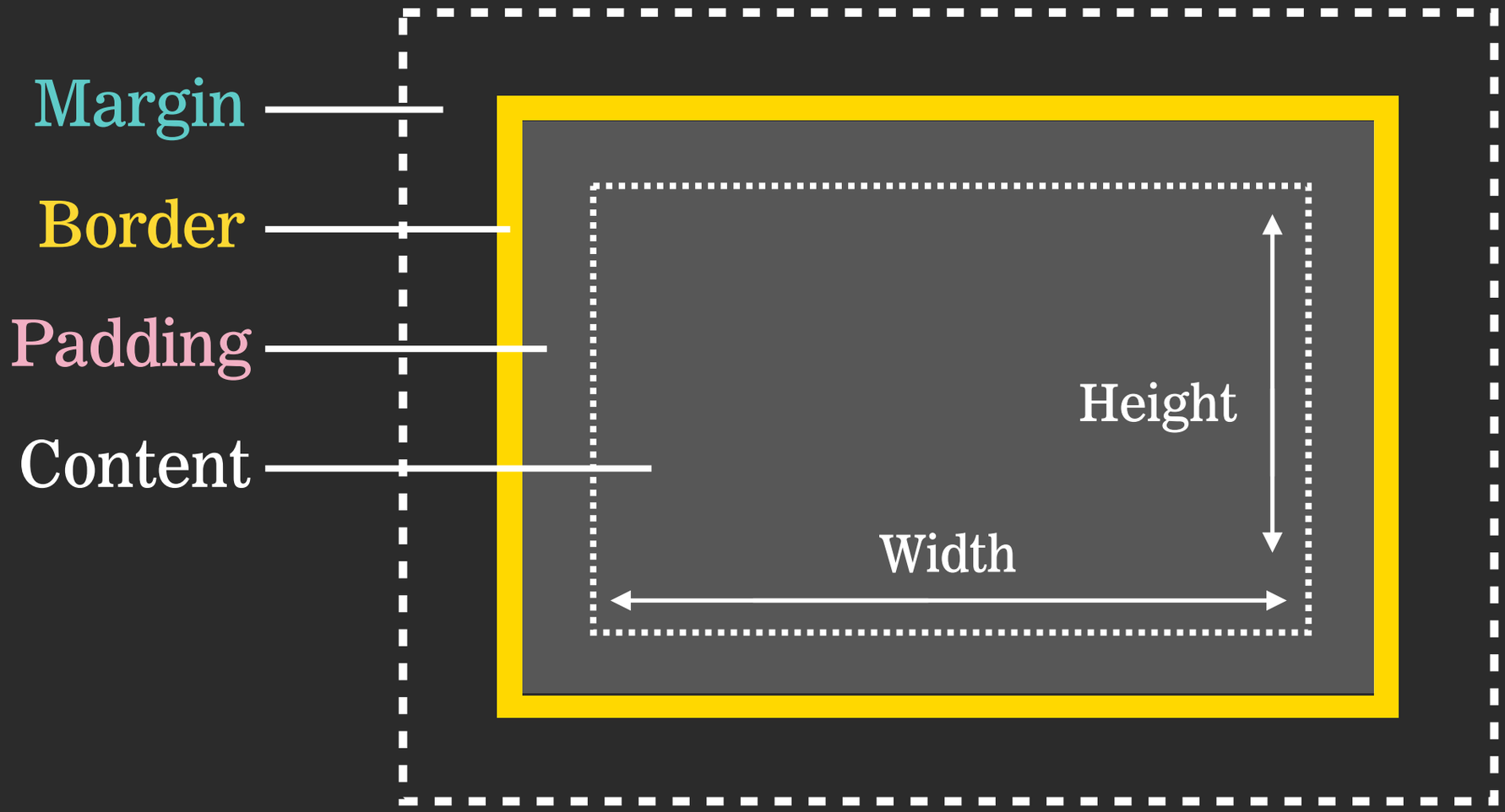
HTML SYNTAX — ATTRIBUTES



CSS SYNTAX

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

CSS — BOX MODEL



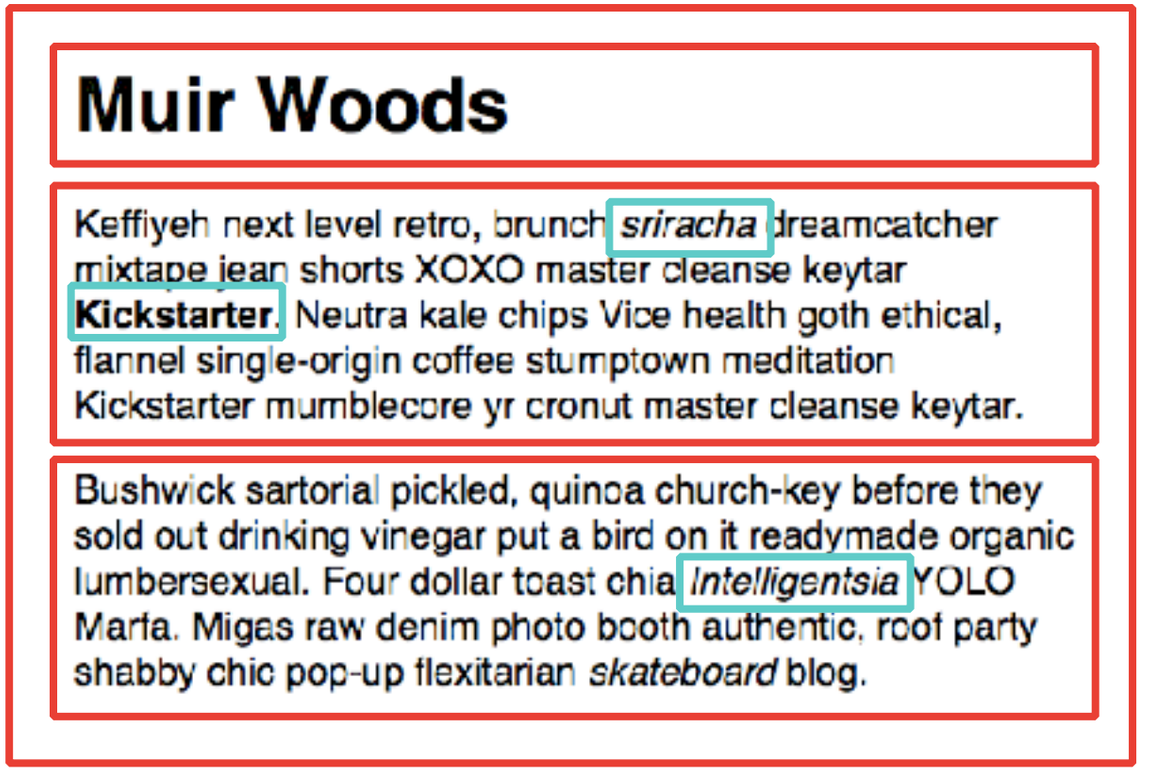
BOX MODEL

REMEMBER THAT THING WE MENTIONED?

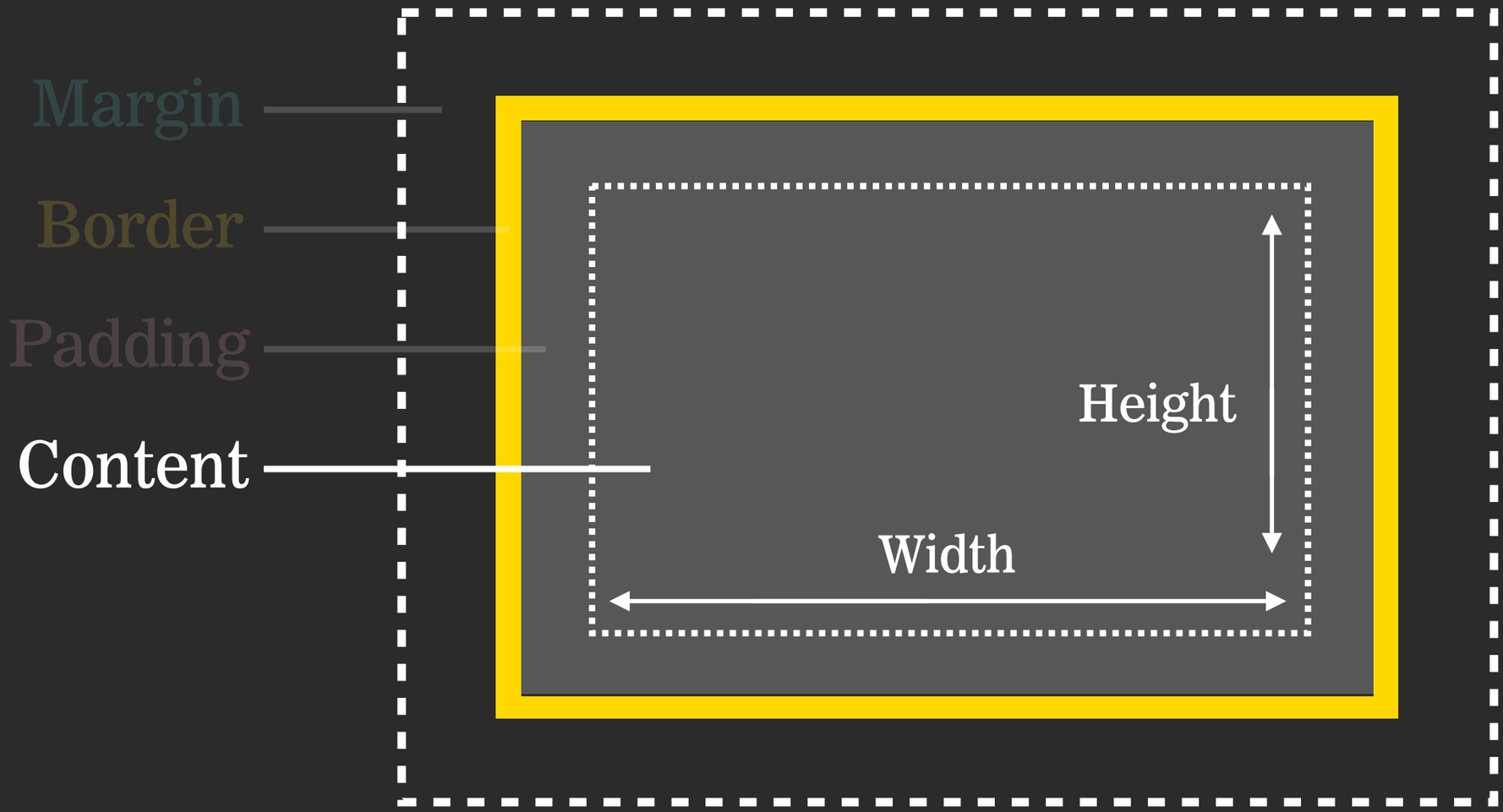
Muir Woods

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

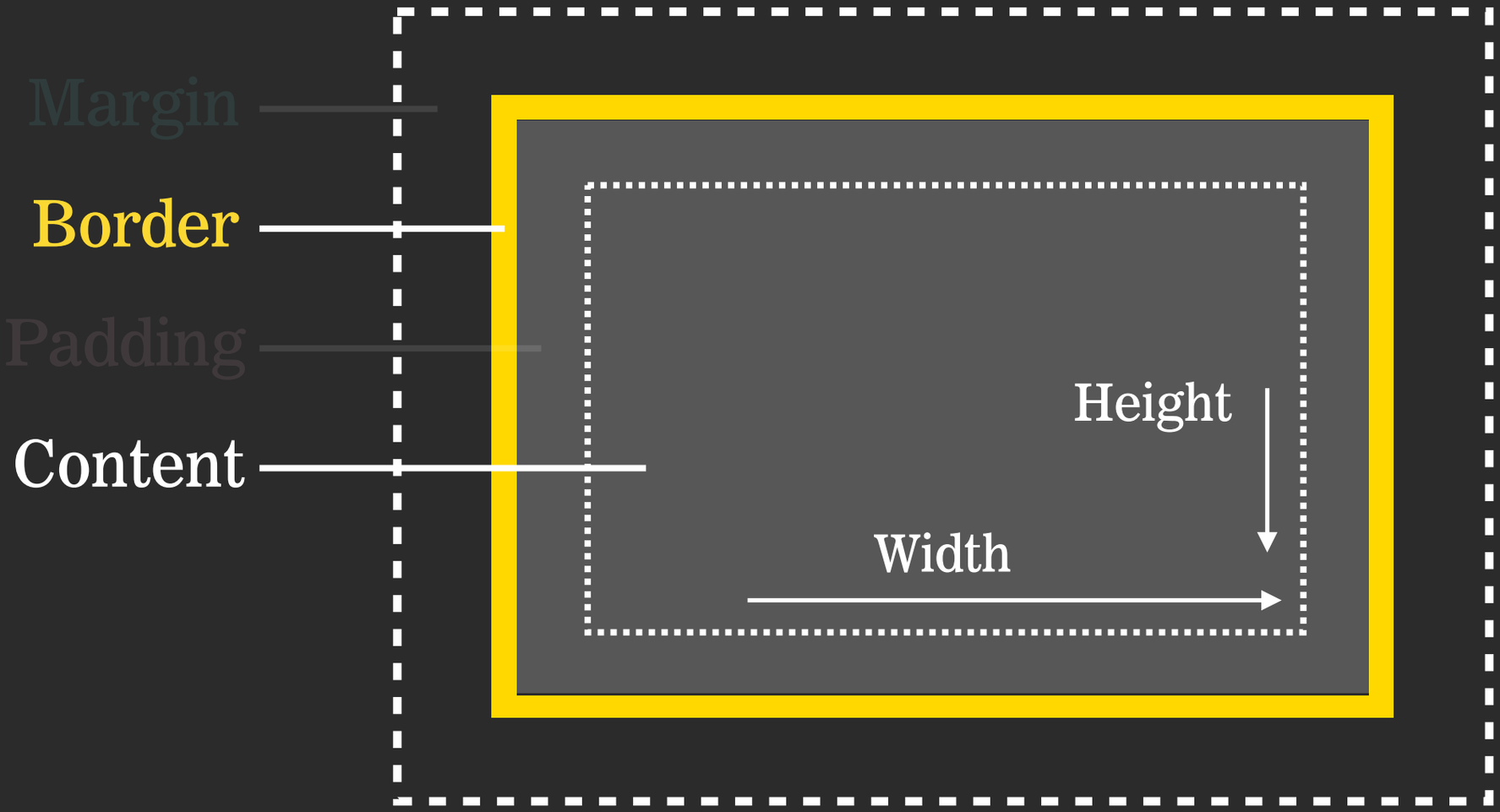
Every box has three properties that can be adjusted to control its appearance:

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

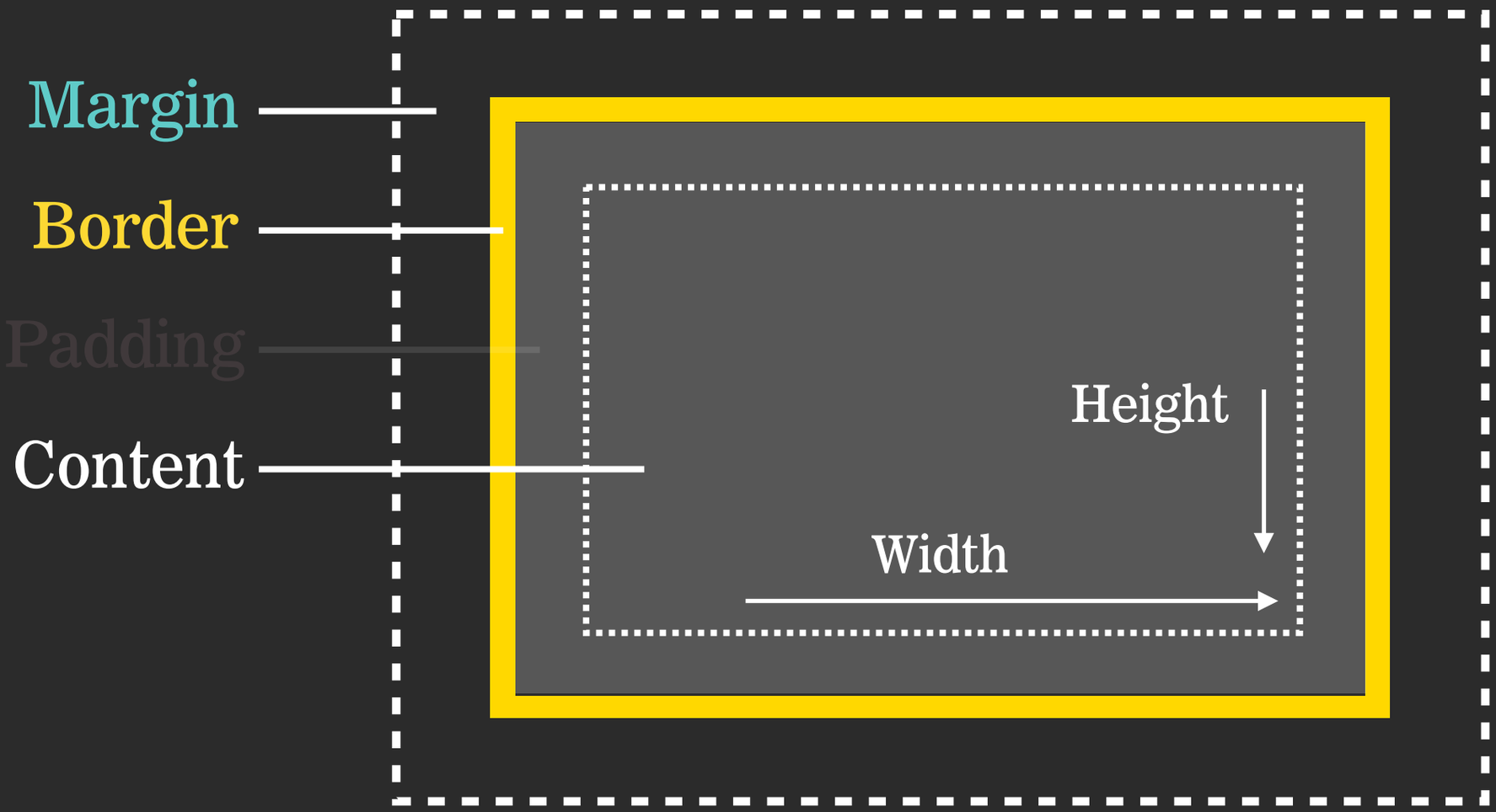
Every box has a border, even if it's not visible or specified to be 0 px wide

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

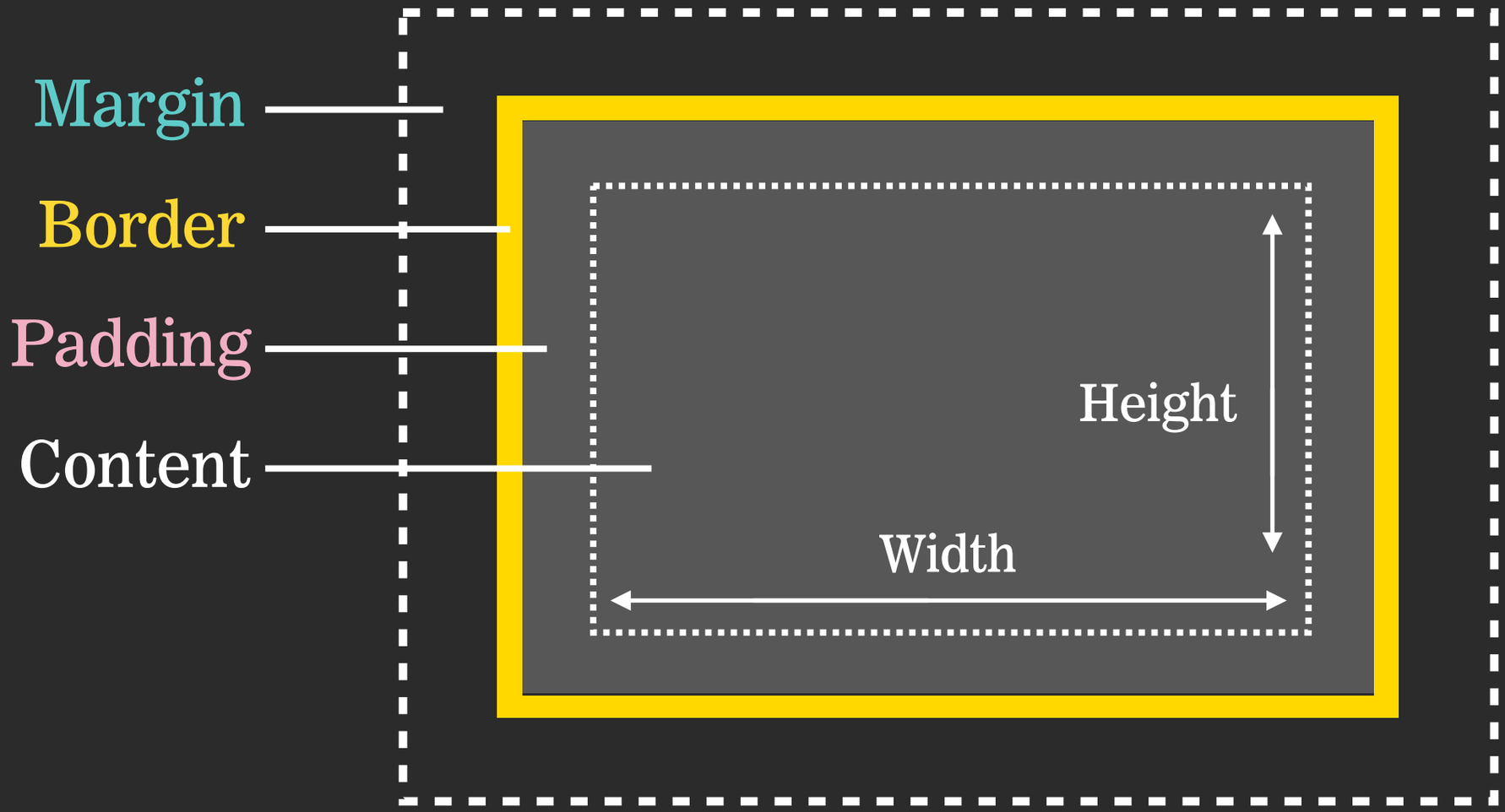
2
MARGIN

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;

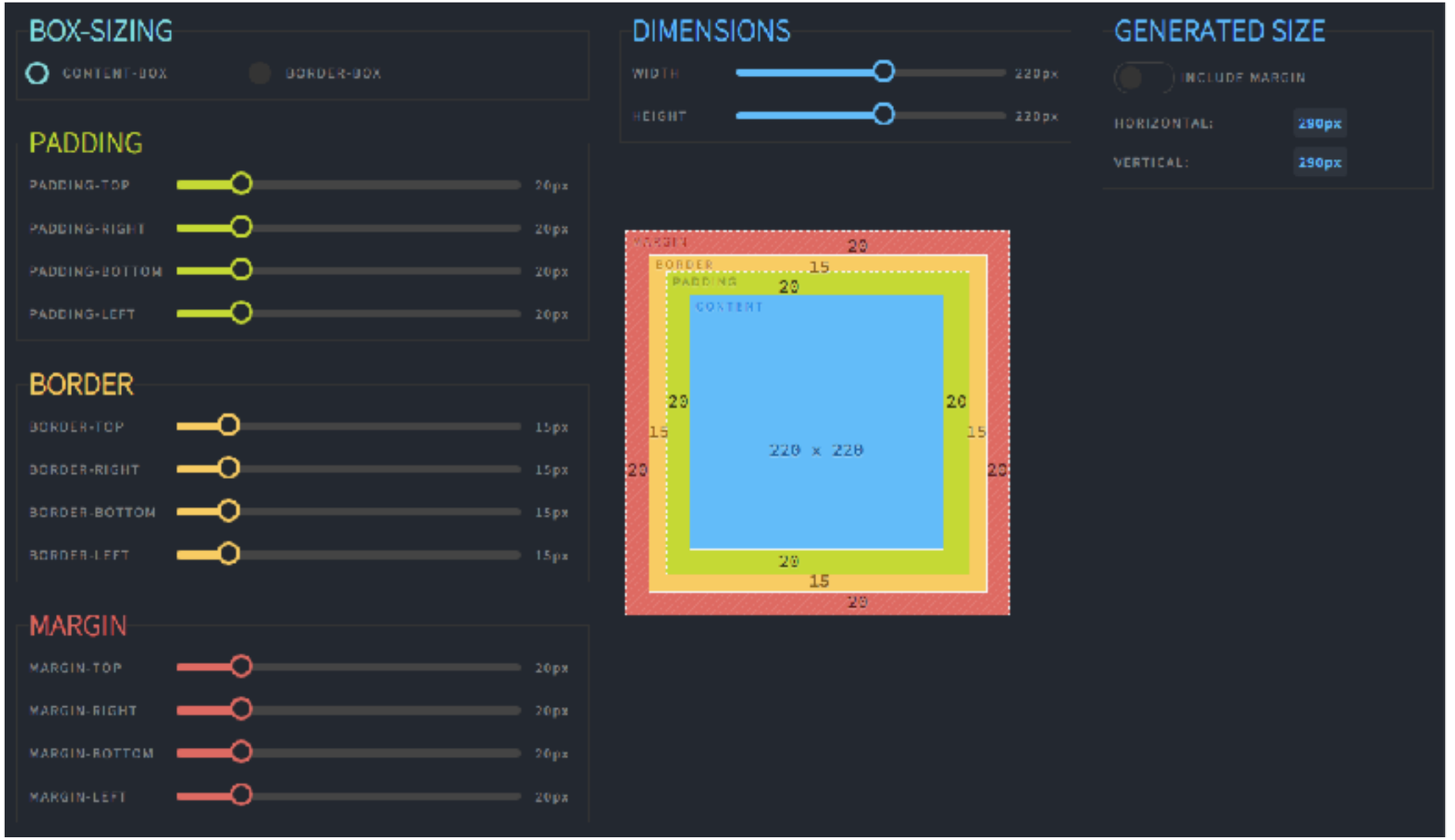
2. Specific side

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

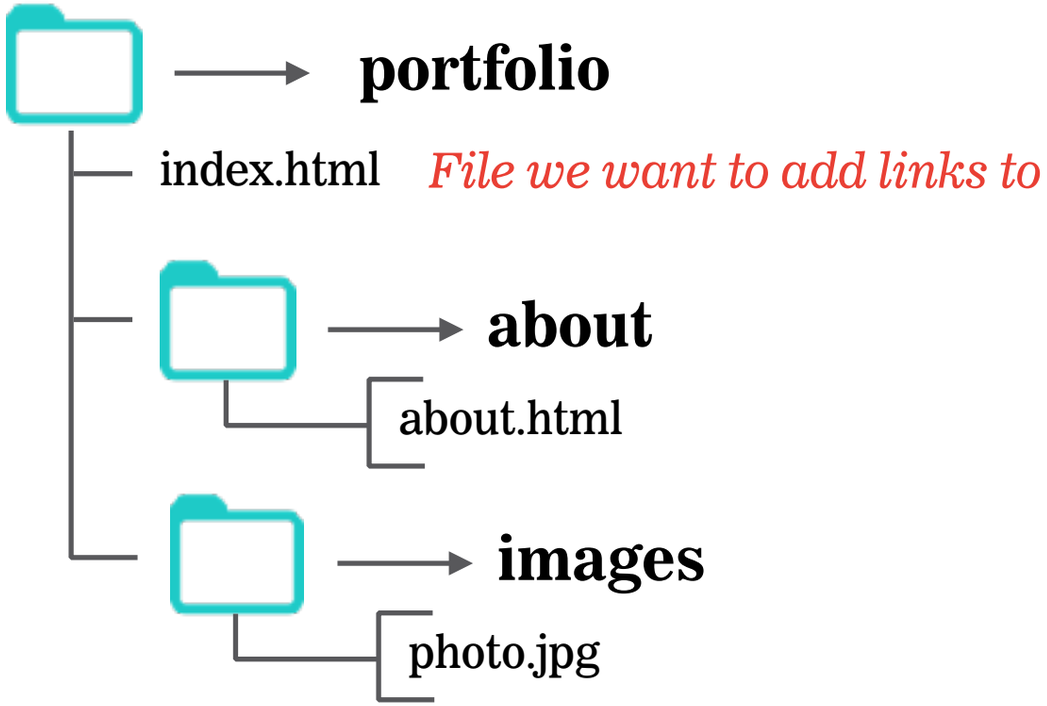
1. Refer back to starter_code > css_practice > index.html
2. Follow the instructions under Part 3

CSS SELECTORS

RELATIVE URLS — SAME FOLDER

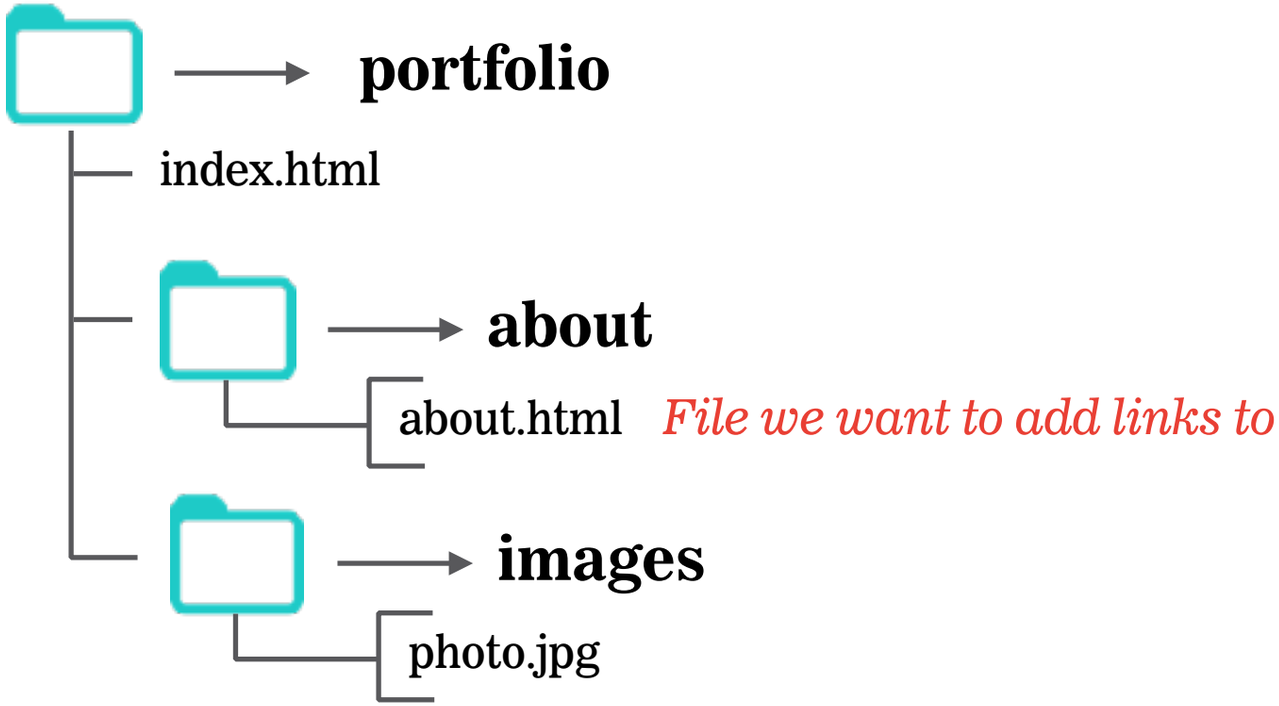
	RELATIVE URL	EXAMPLE
SAME FOLDER	File name	<a href="info.html">Info</a>
		<img src="photo.jpg" alt="">

RELATIVE URLS — CHILD FOLDER



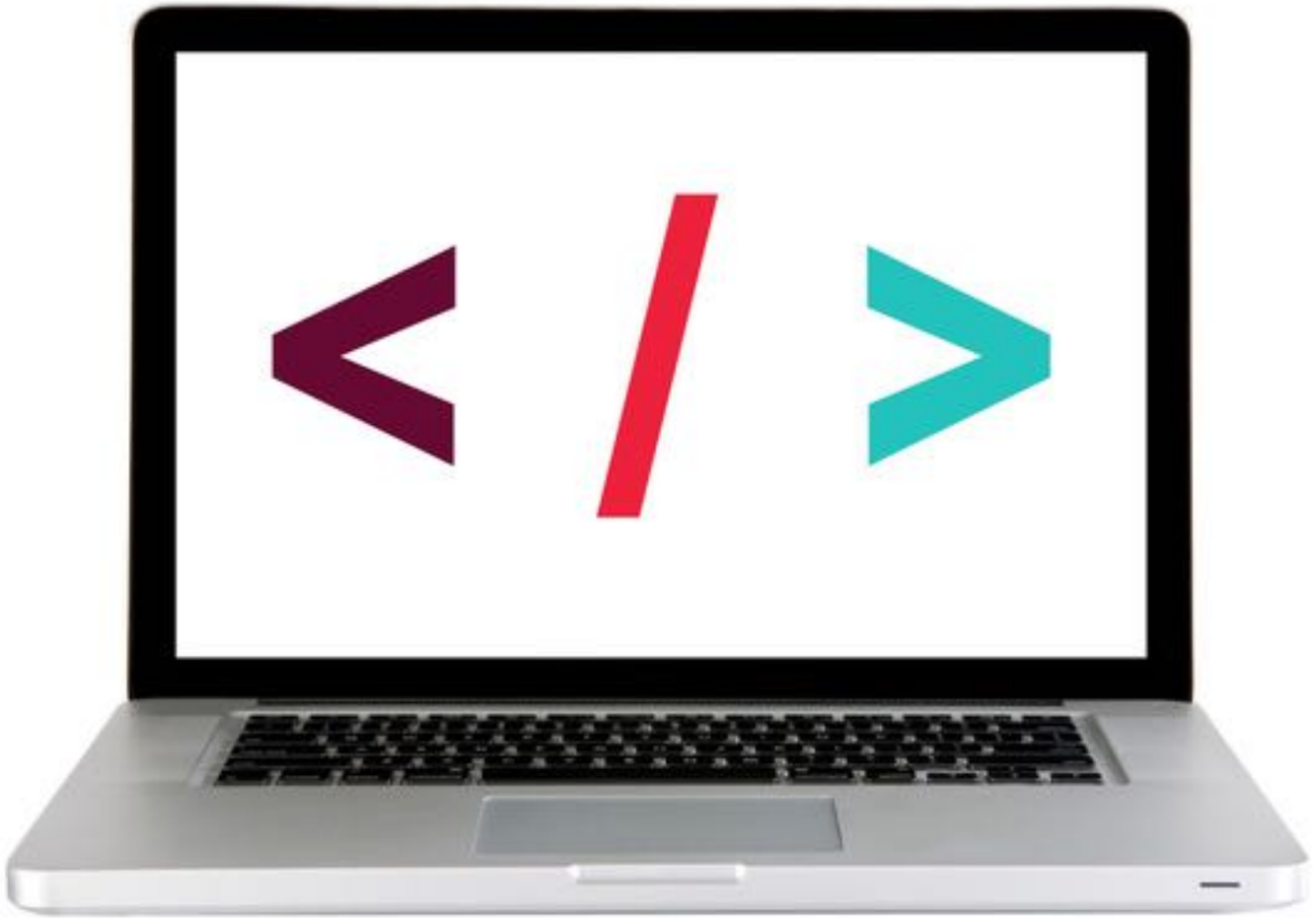
	DESCRIPTION	EXAMPLE
CHILD	Name of child folder + / + file name	<img src="images/photo.jpg">
		<a href="about/about.html">About</a>

RELATIVE URLS — CHILD FOLDER



	DESCRIPTION	EXAMPLE
PARENT	../ + path	<a href="../index.html">Home</a>
		<img src="../images/photo.jpg">

LET'S TAKE A CLOSER LOOK



ACTIVITY

EXERCISE

KEY OBJECTIVE

- Practice adding relative URLs to a project

KEY OBJECTIVE

- Starter code > Background image

TIMING

4 min

1. With a partner add a background-image to the style.css file. You'll want to use the background-image property along with the background-size property.
2. Test in browser!

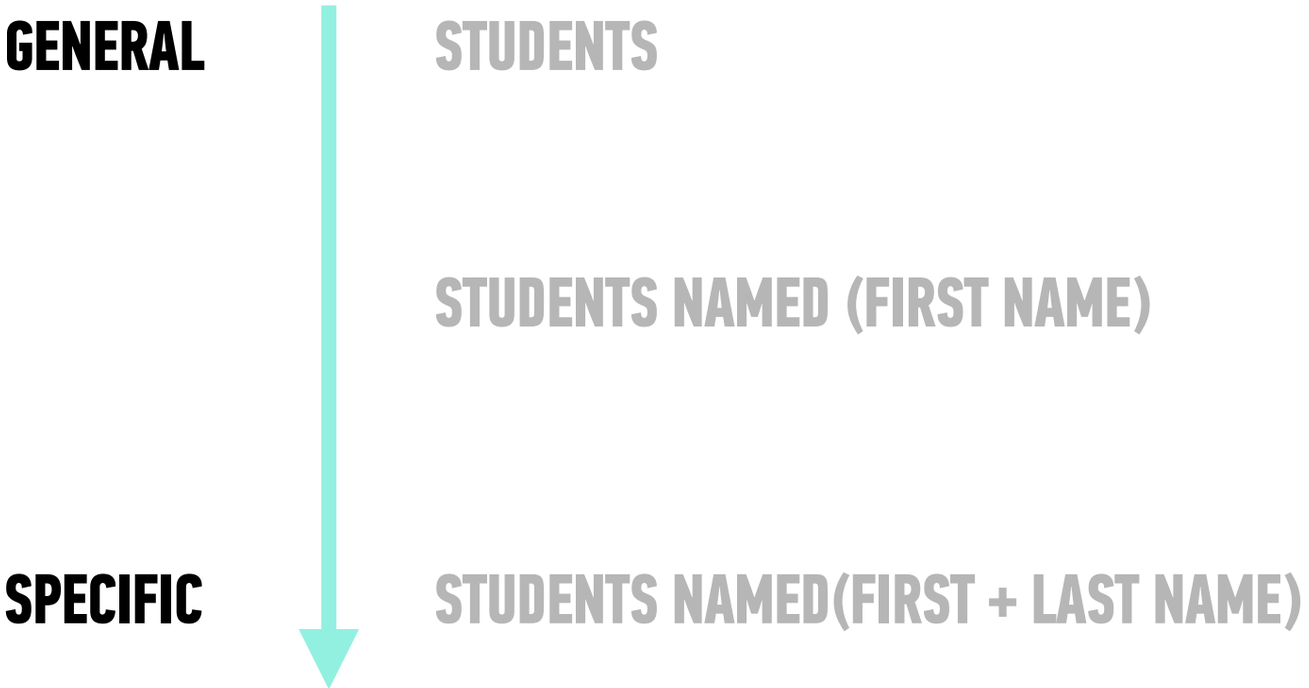
RELATIVE URLS

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

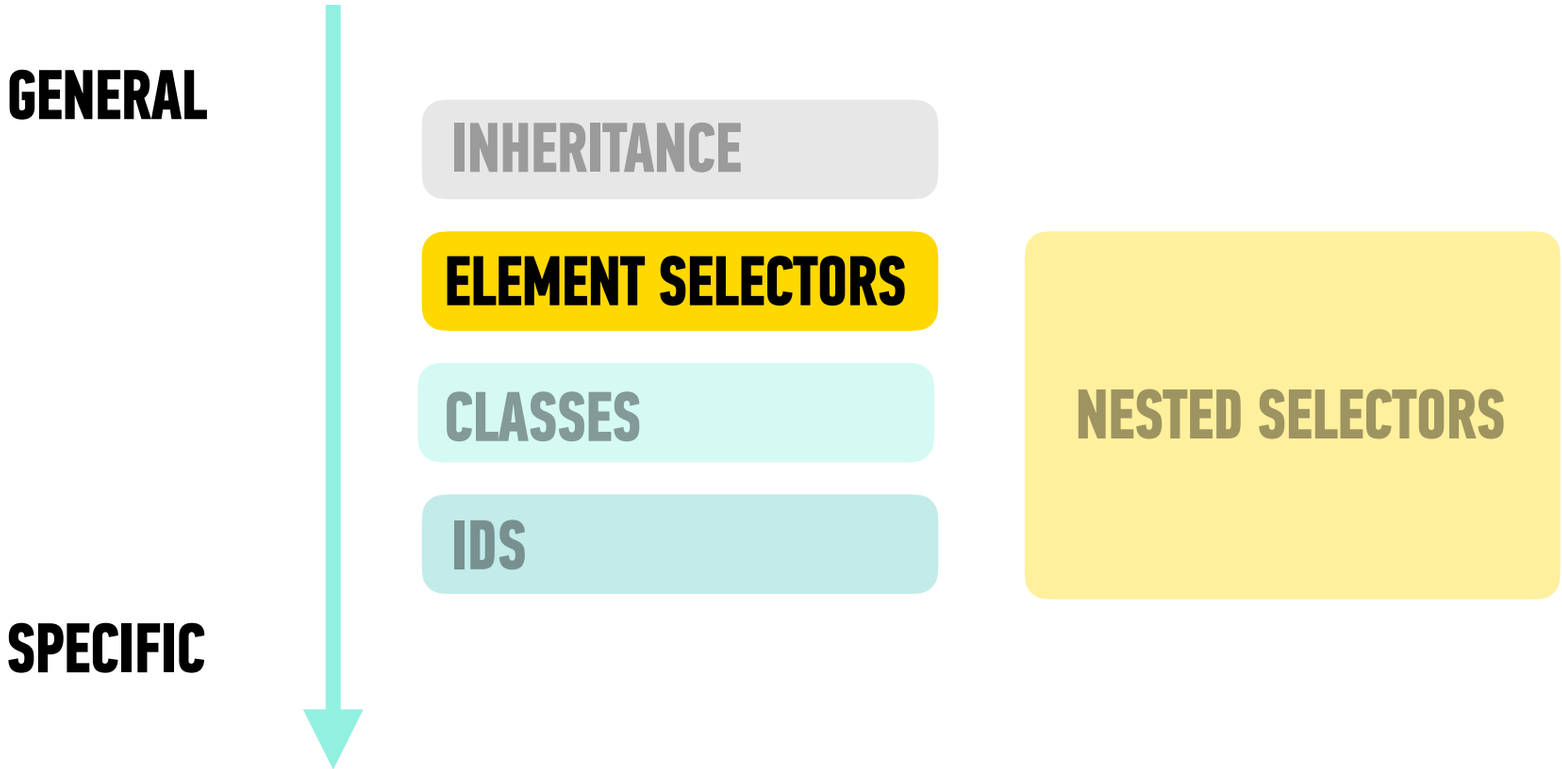
Note that ../ means to go up one directory, and can be used repeatedly:
../../ would go up two directories.

LET'S CHAT ABOUT THE CASCADING NATURE OF CSS

MORE ABOUT CASCADING



MORE ABOUT CASCADING — GENERAL TO MORE SPECIFIC



INHERITANCE

INHERITANCE — SETTING BASE STYLES

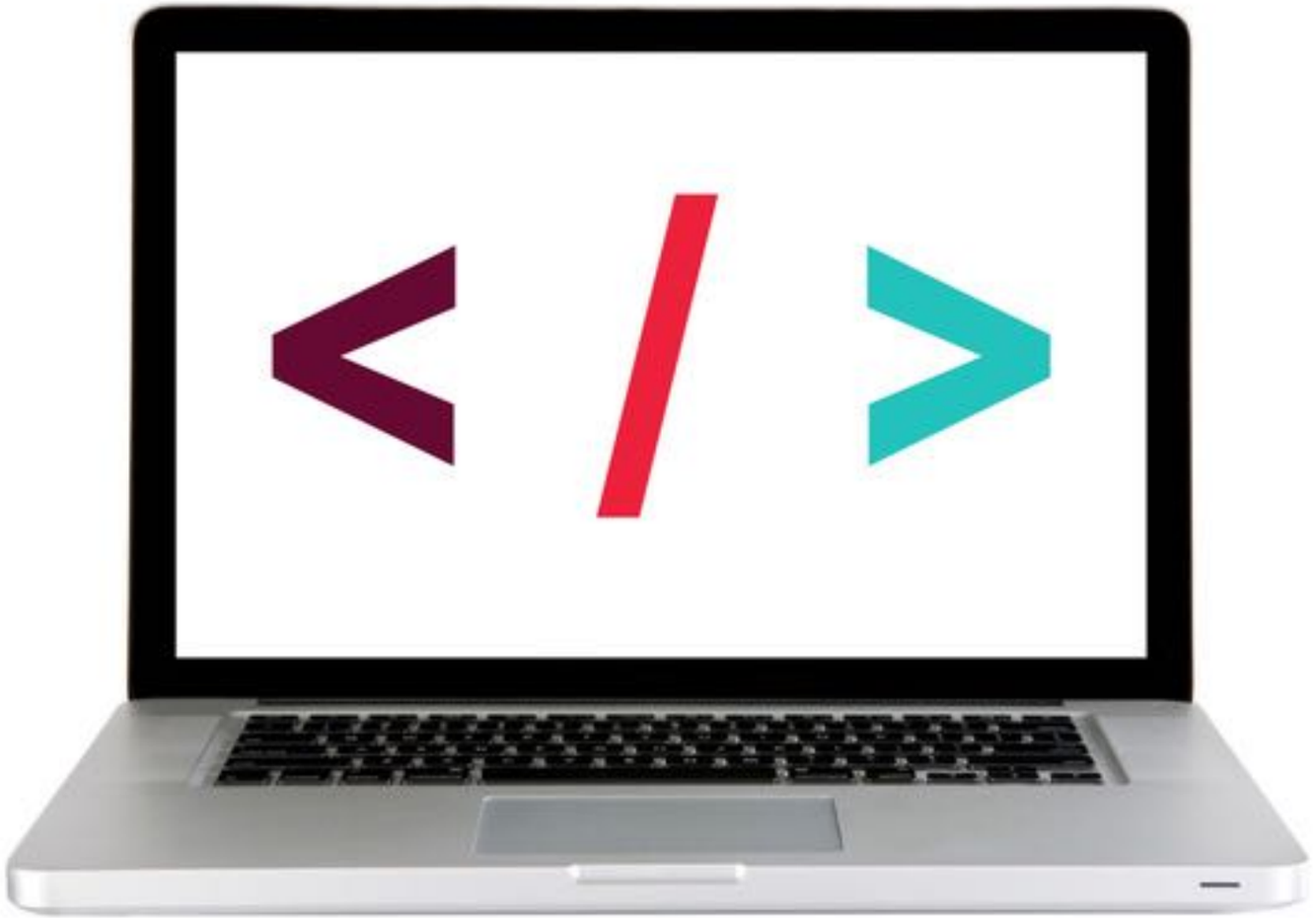
INHERITANCE — SETTING BASE STYLES

- 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:

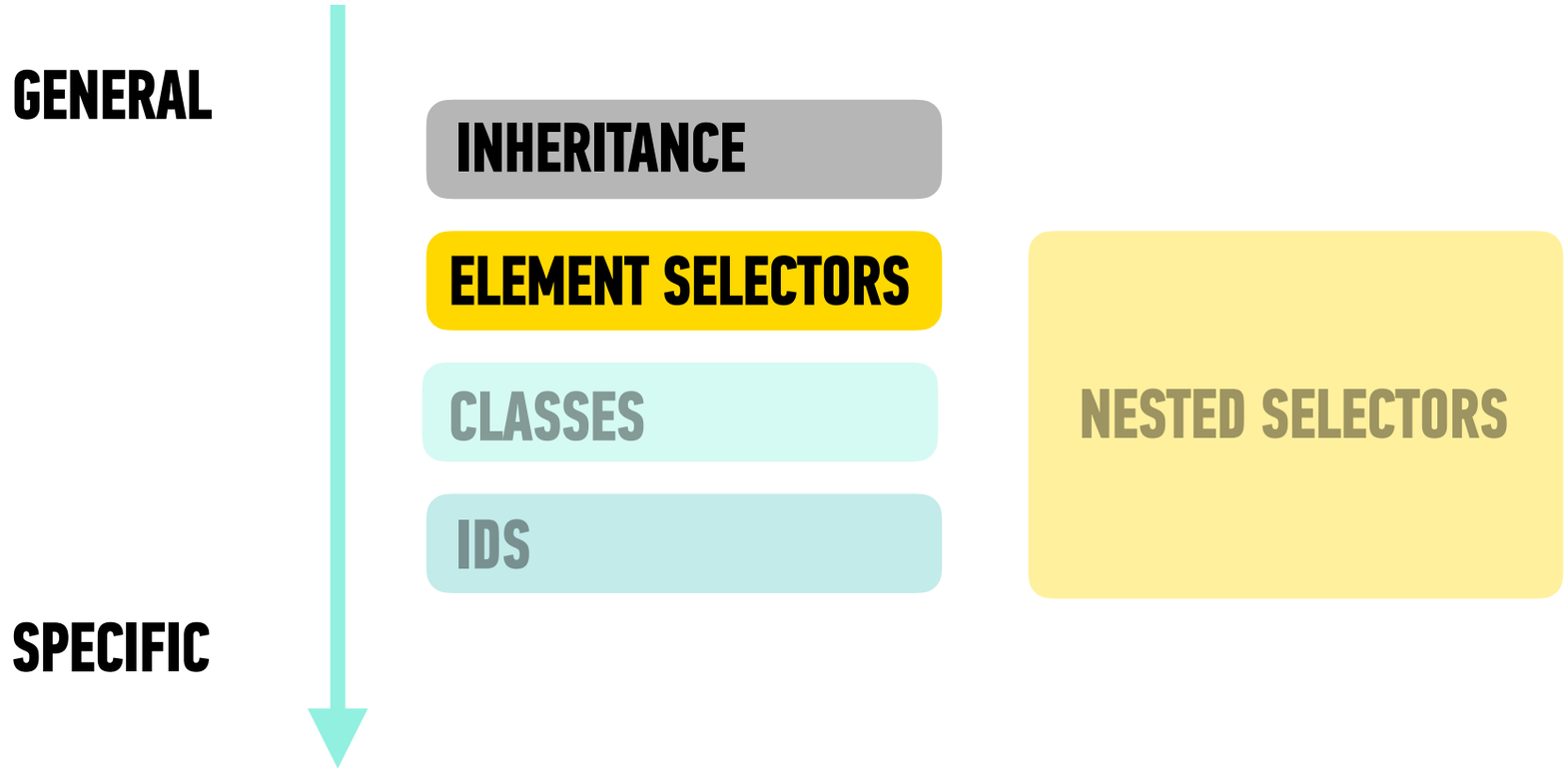
- color
- font-family
- font-size
- font-style
- font-weight
- letter-spacing
- line-height
- list-style
- text-align
- text-indent
- text-transform
- word-spacing

LET'S TAKE A CLOSER LOOK



MORE ABOUT CASCADING — GENERAL TO MORE SPECIFIC

- CSS rules cascade downward until they are canceled out by another rule.



ACTIVITY

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

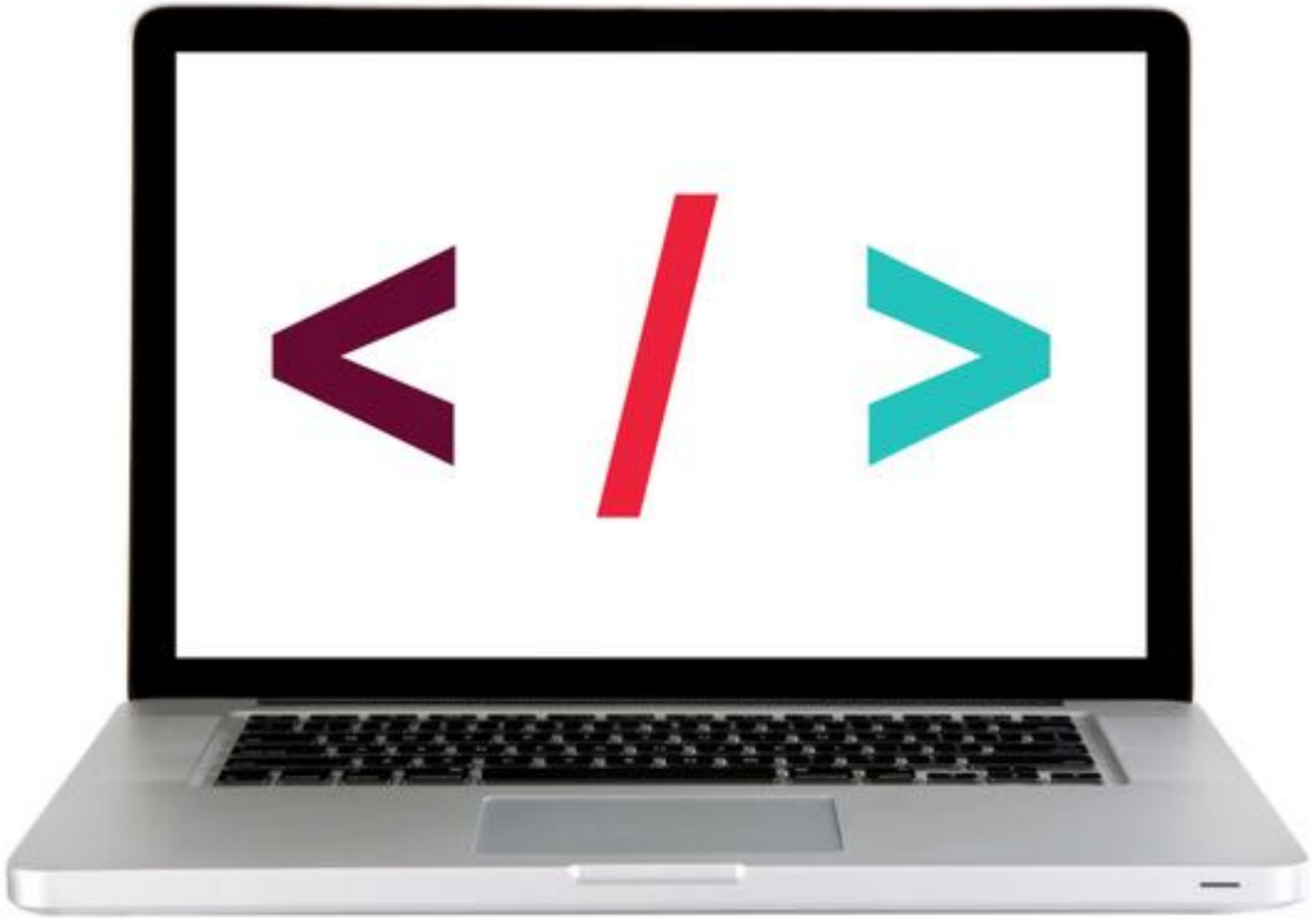
- Turn and Talk

TASKS

1 min	1. Turn and talk to a partner
2 min	2. Pairs will share with the class

CLASSES AND IDS

LET'S TAKE A CLOSER LOOK



TARGETING SPECIFIC ELEMENTS

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

TARGETING SPECIFIC ELEMENTS

CLASSES

- Classes are used to group elements together
- Elements can have multiple classes

```
<li class="emphasis">Content</li>
```

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

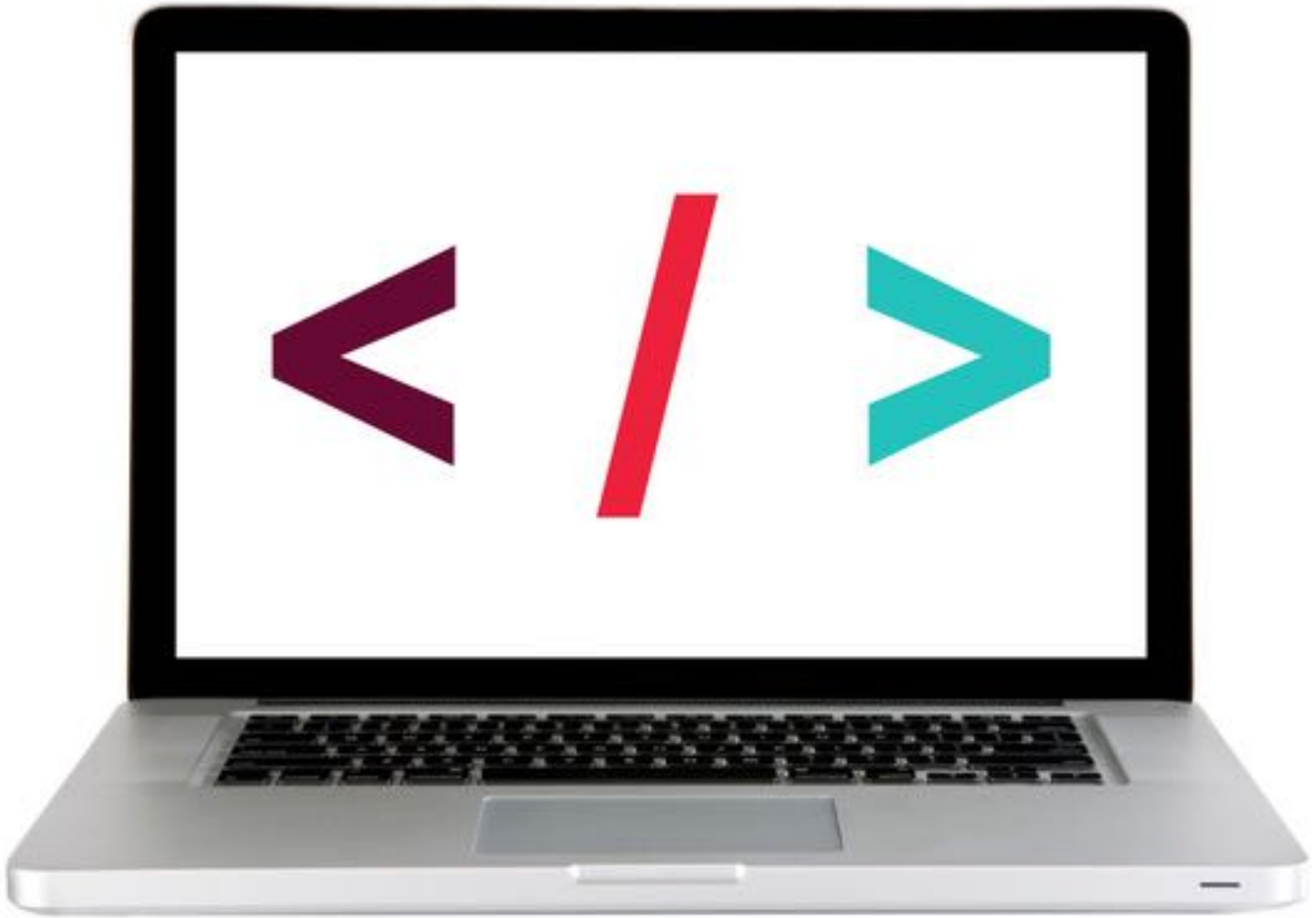
IDS

- 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

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

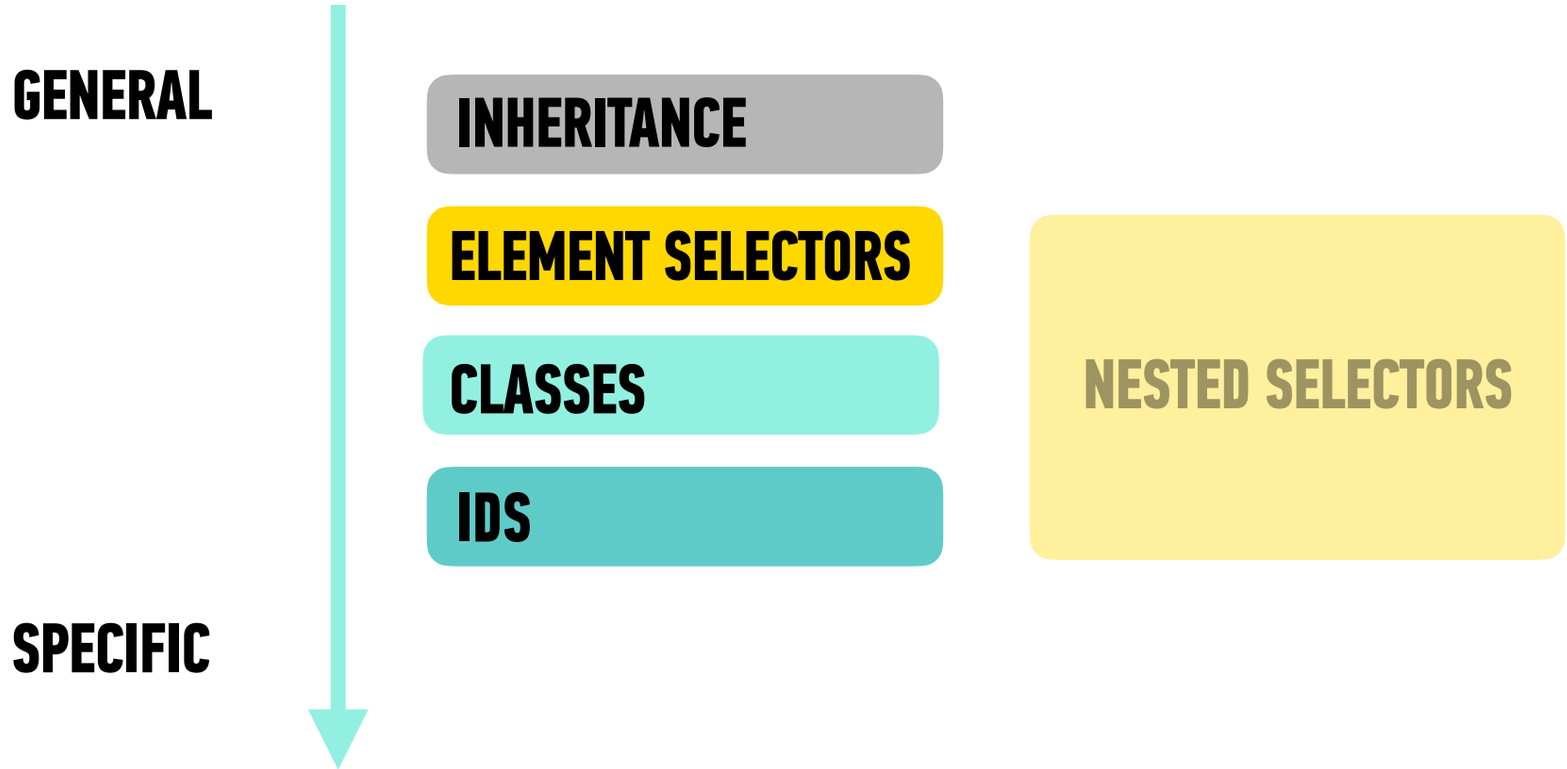
```
#main-nav {
  text-align: center;
}
```

LET'S TAKE A CLOSER LOOK



MORE ABOUT CASCADING — GENERAL TO MORE SPECIFIC

- CSS rules cascade downward until they are canceled out by another rule.



ACTIVITY

EXERCISE

LOCATION

- starter_code folder > classes_and_ids folder

KEY OBJECTIVE

- Use classes and IDs to target elements

TIMING

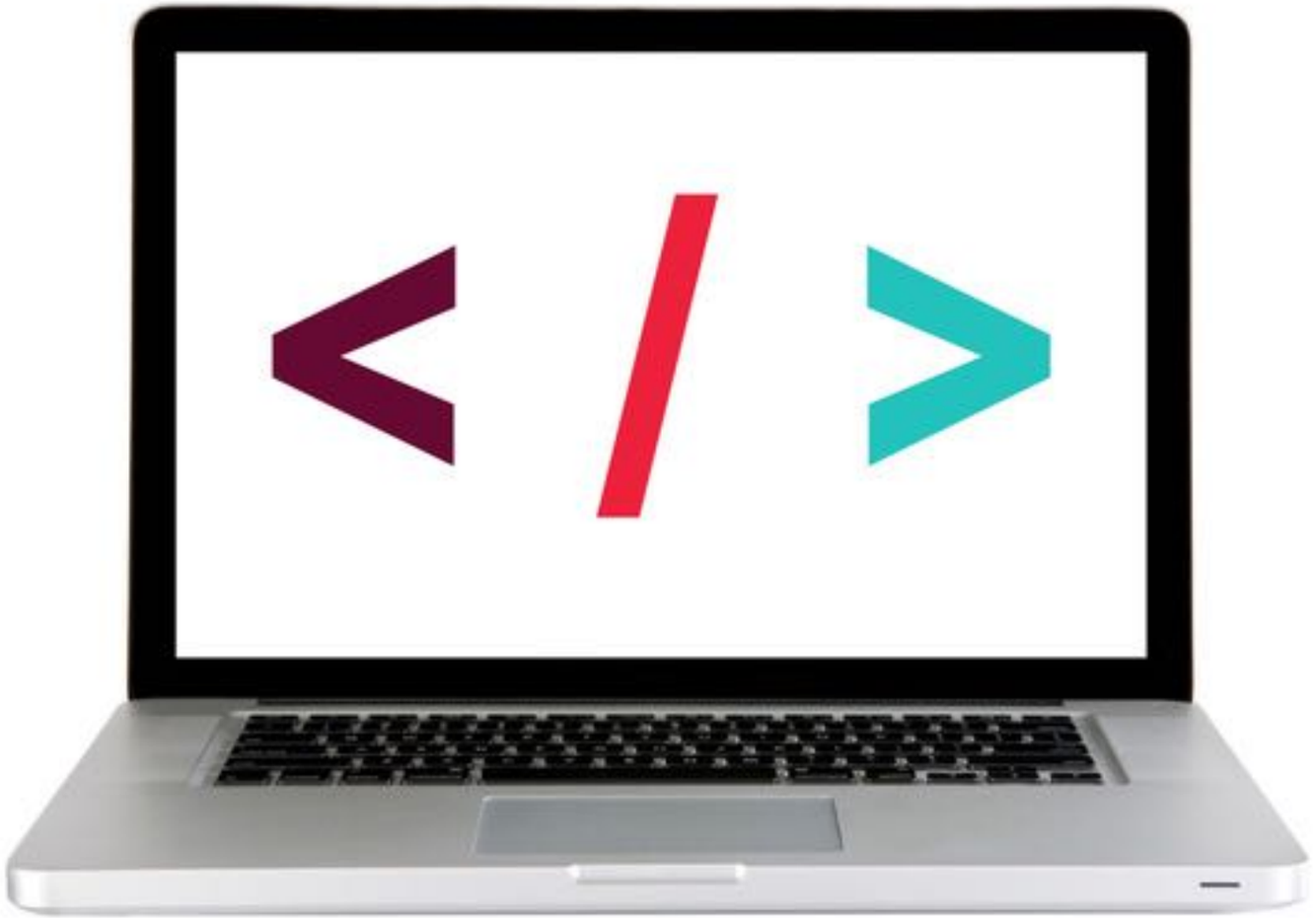
5 min 1. Follow the instructions under 'Now You Try'

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.

NESTED SELECTORS

LET'S TAKE A CLOSER LOOK



SELECTORS — MOST COMMON

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	Select multiple elements	h1, h2 {}

SELECTORS

Which element here will be italic?

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

SELECTORS

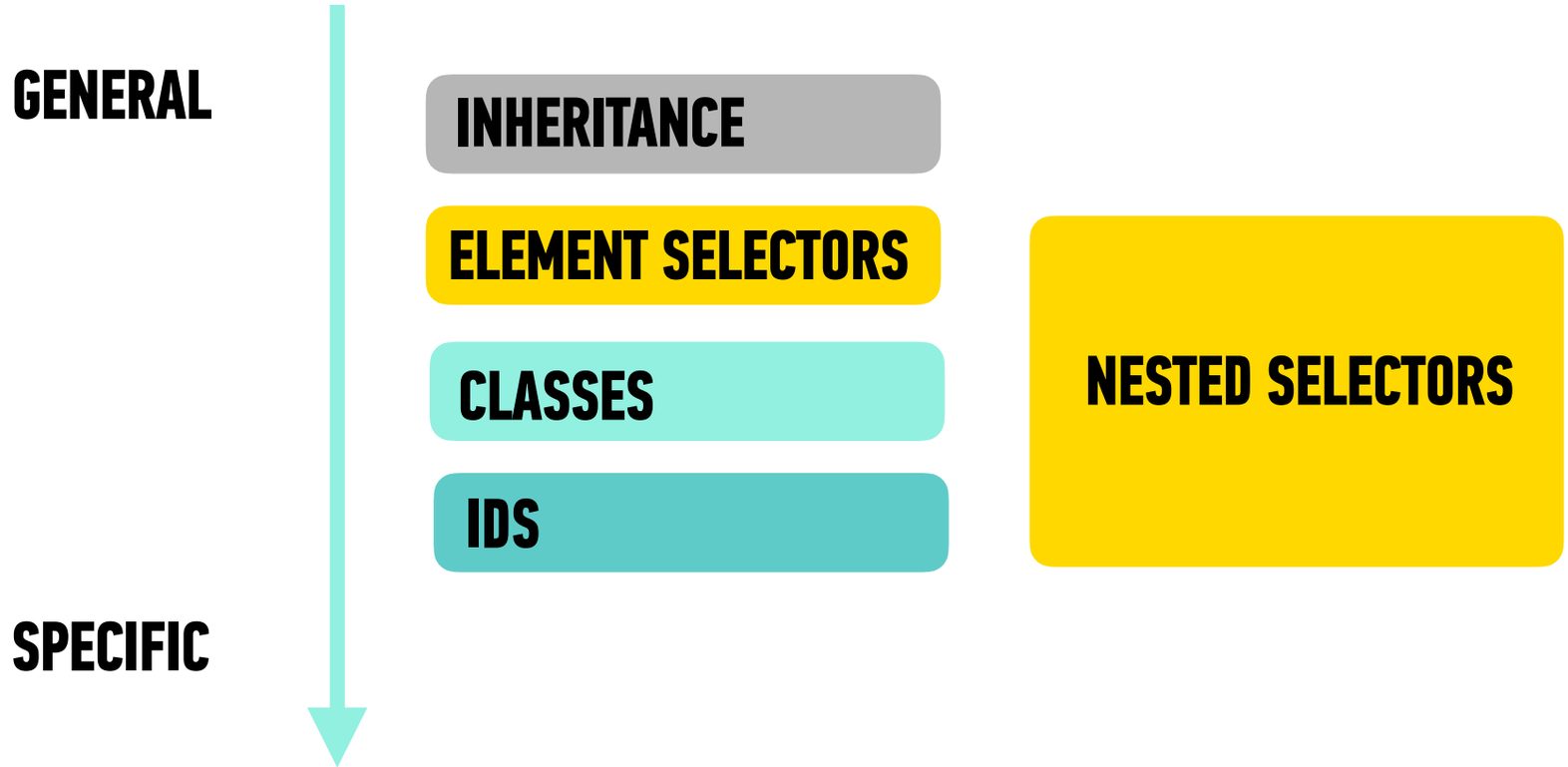
The last element in the selector string is always the one that we are styling

NESTED SELECTORS

```
.error p a {
  color: red;
}
```

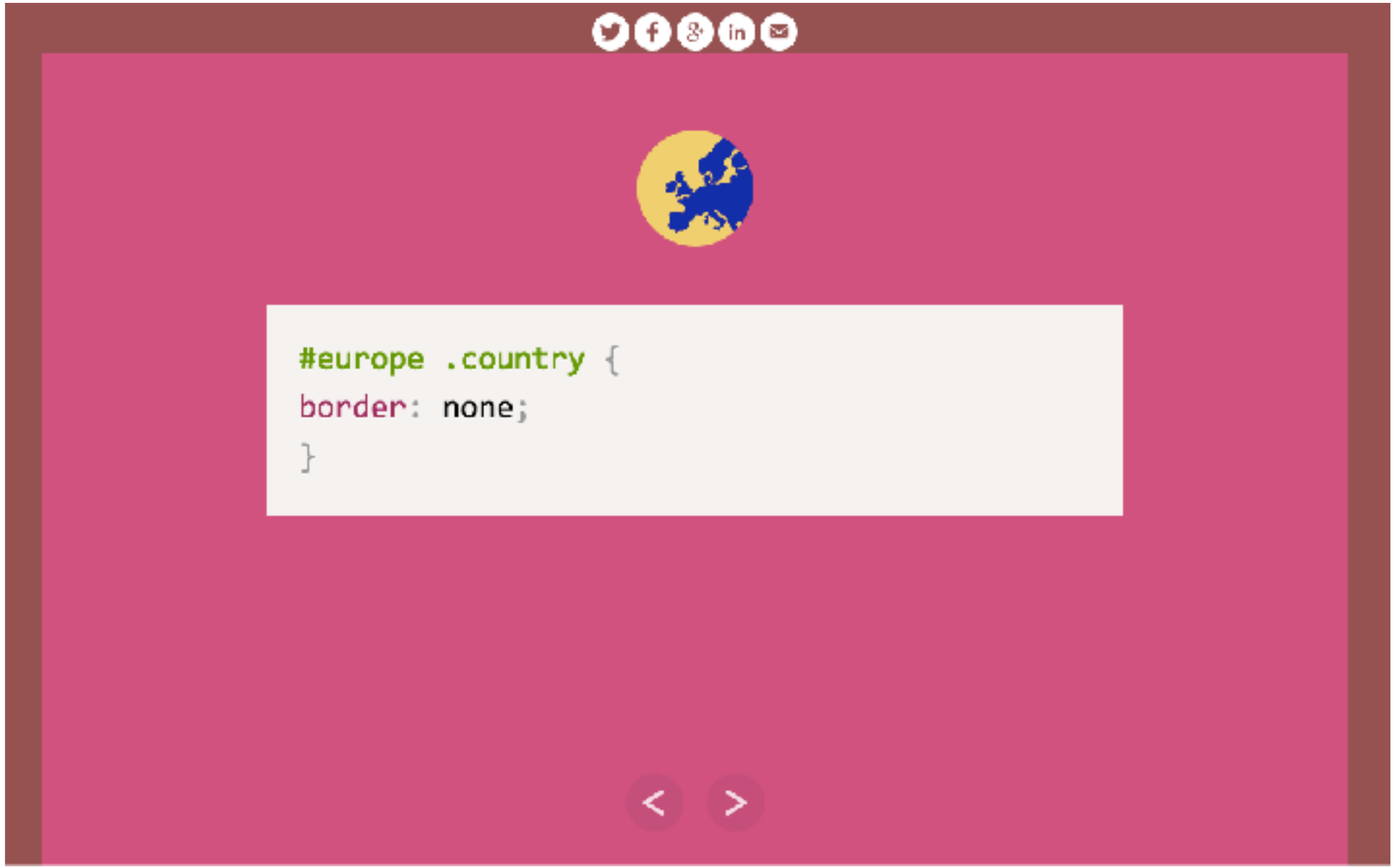
MORE ABOUT CASCADING

- CSS rules cascade downward until they are canceled out by another rule.



CSS JOKES

CSS PUNS



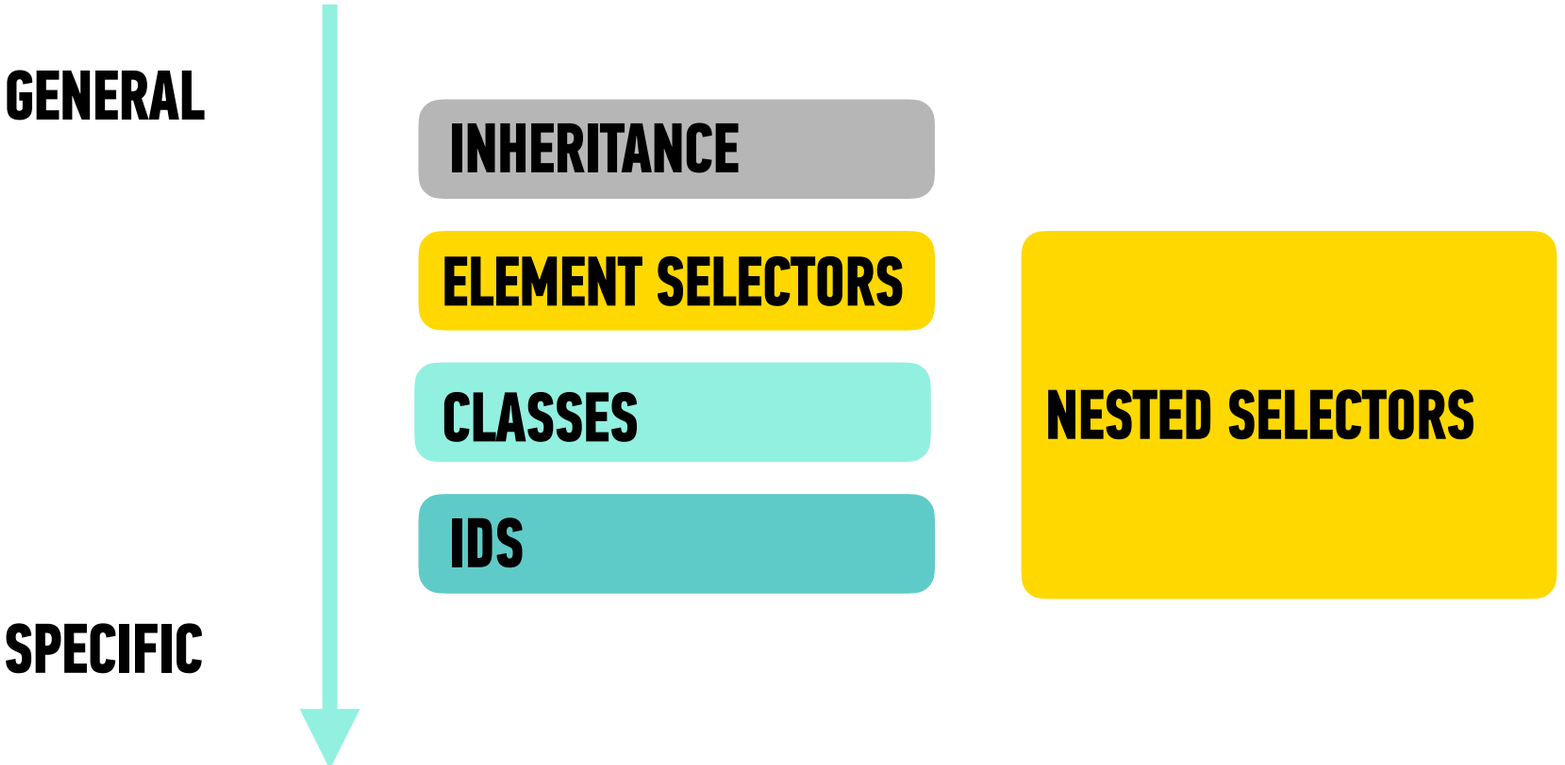
CSS PUNS

SPECIFICITY GAME!!!

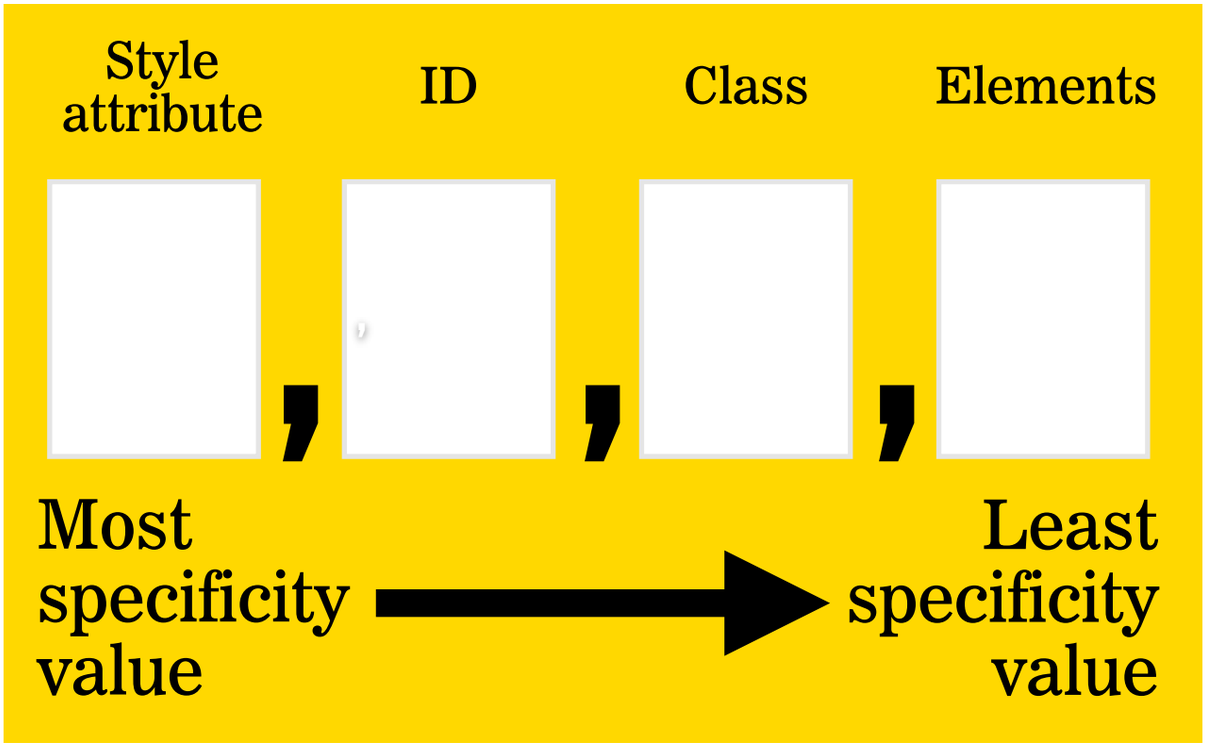
CONFLICT!! — WHEN TWO RULES APPLY TO THE SAME ELEMENT

SPECIFICITY:

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



THE SPECIFICITY GAME!



CSS CASCADING — THE SPECIFICITY GAME

HTML:

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

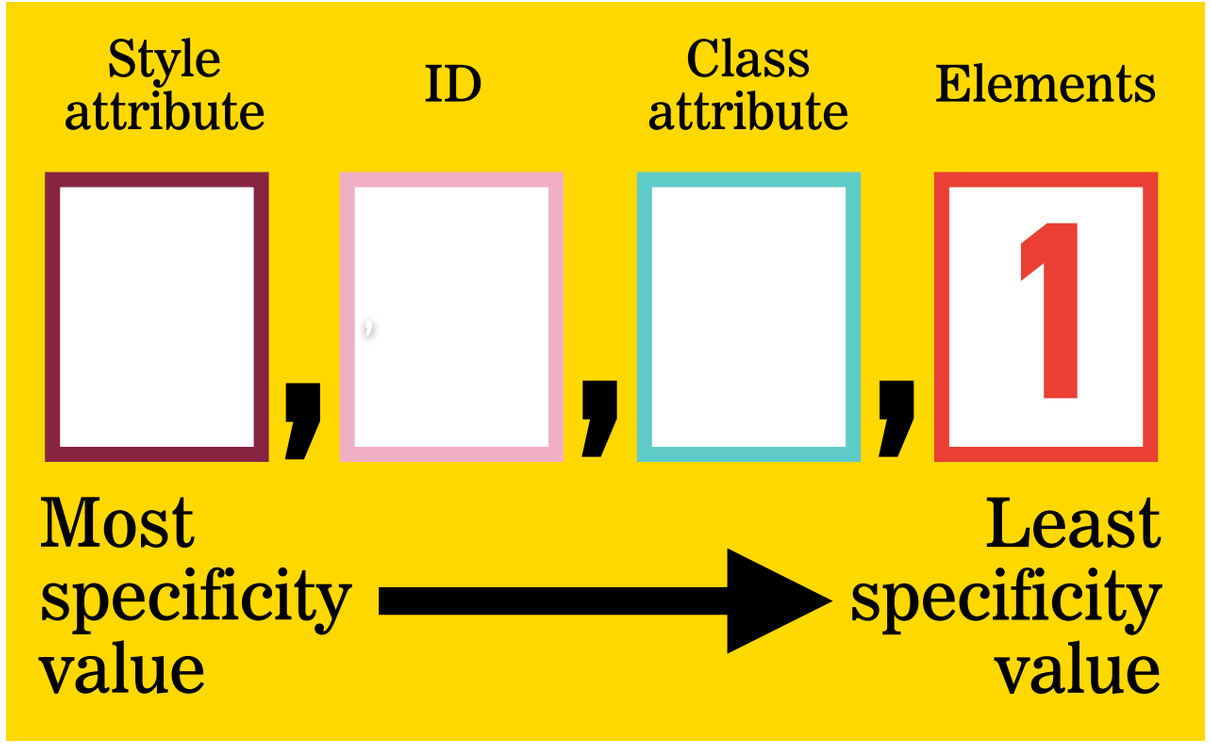
CSS:

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

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

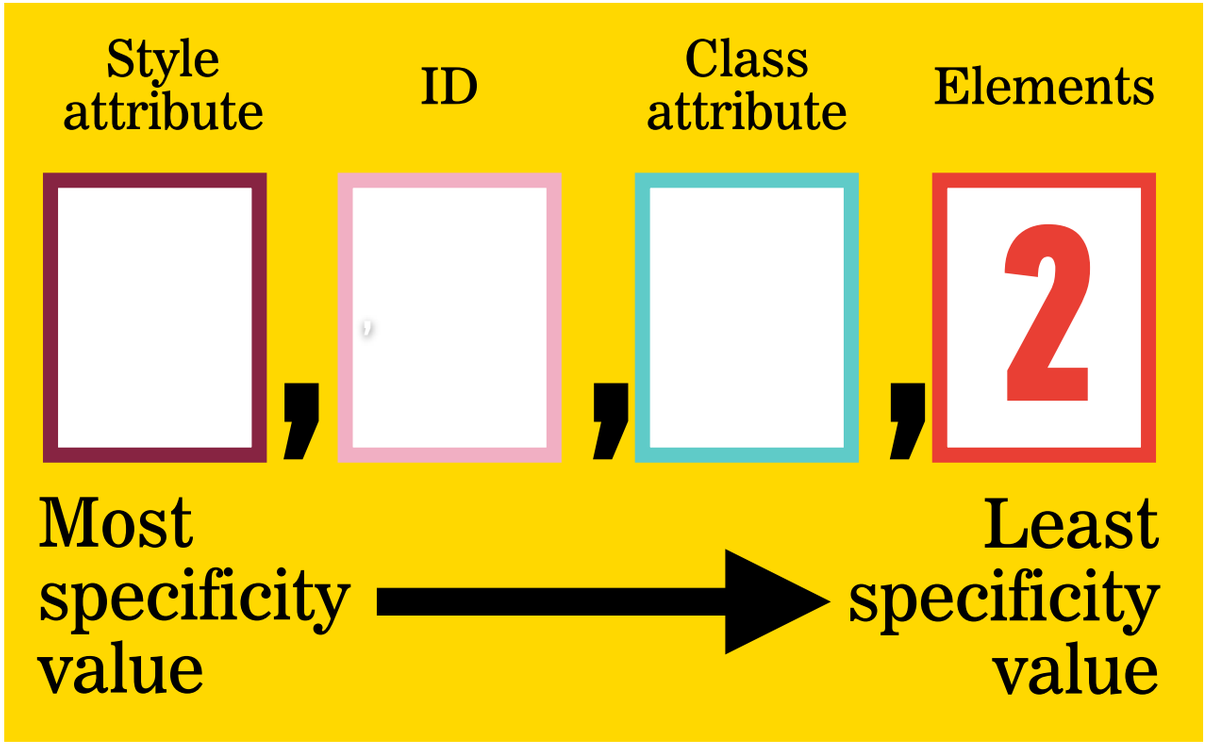
SPECIFICITY

WINNER!

a



p a



CSS CASCADING — THE SPECIFICITY GAME

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

HTML:

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

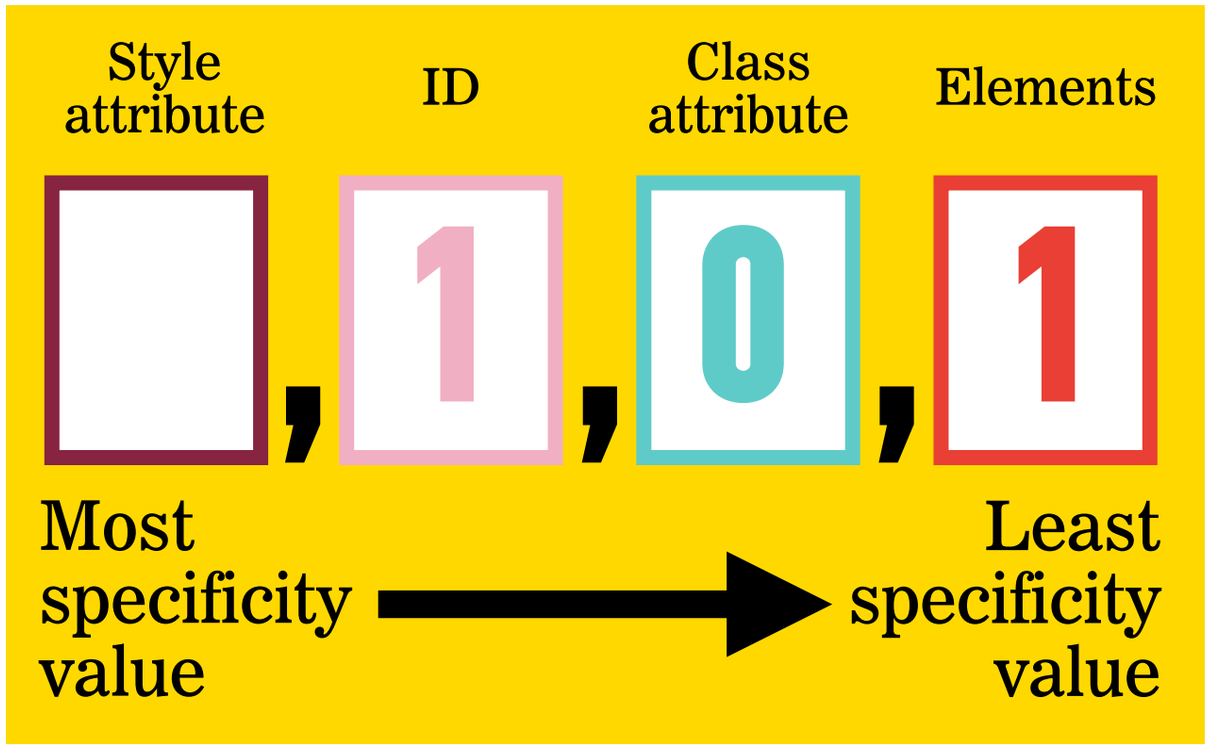
CSS:

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

.home {
  color: blue;
}
```

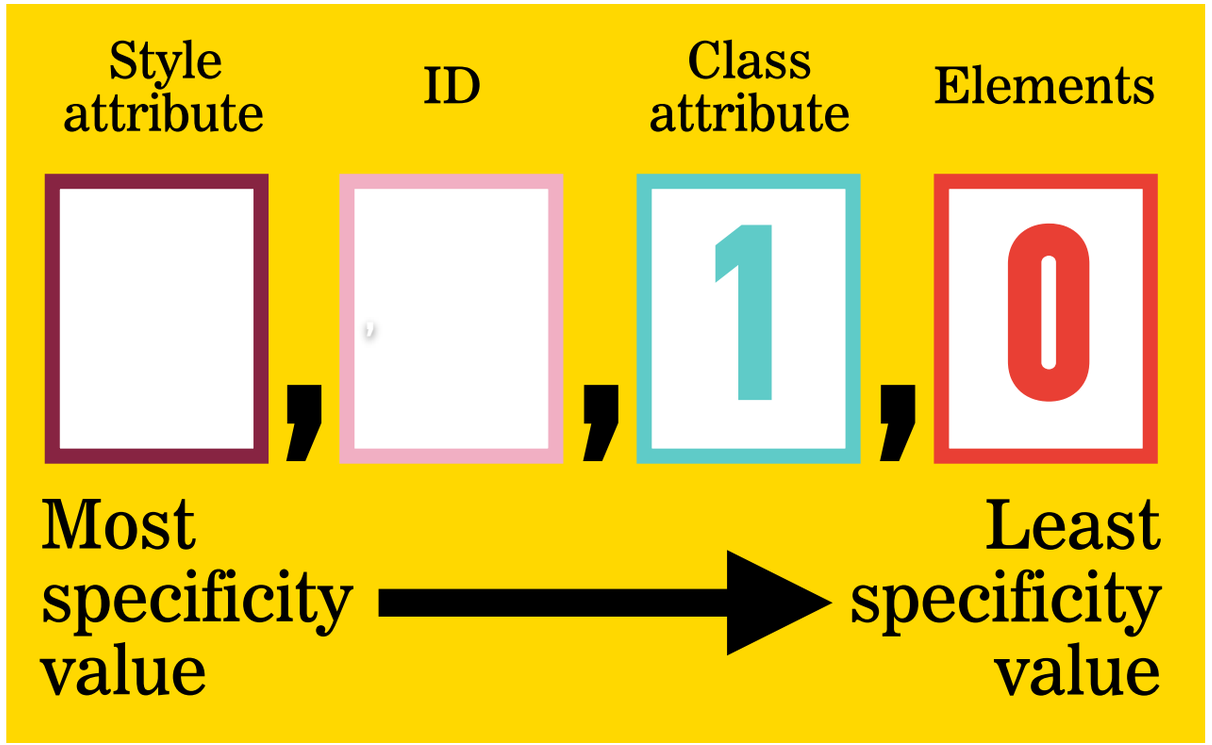
CSS CASCADING

WINNER!

#main-nav a



.home



CSS CASCADING — THE SPECIFICITY GAME

HTML:

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

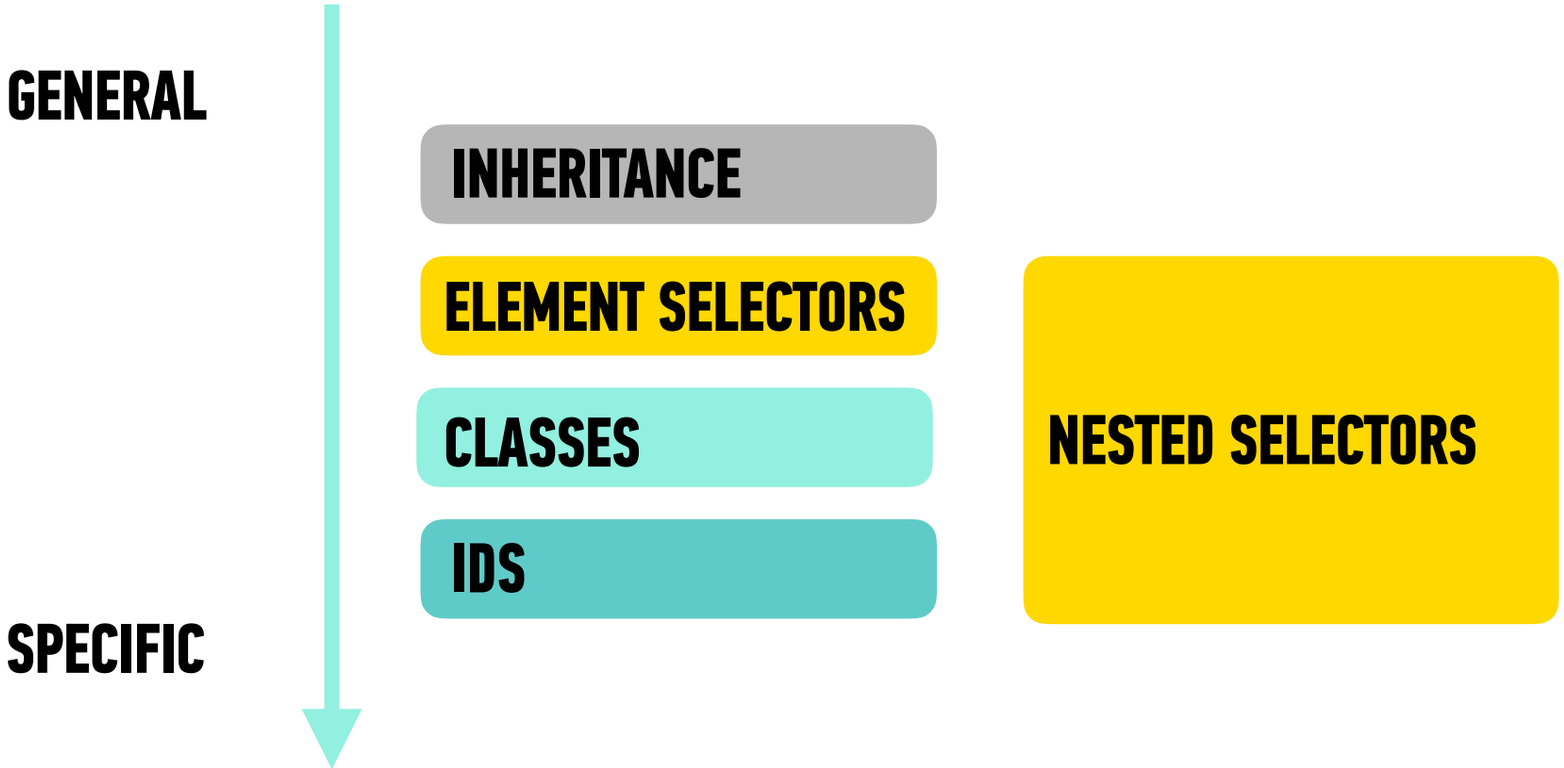
CSS:

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

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

CSS CASCADING — THE SPECIFICITY GAME

SPECIFICITY:

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



LAST RULE

CSS CASCADING — THE SPECIFICITY GAME

HTML:

```
<li>
  <p>Visit this <a href="">Cool Site</a></p>
</li>
```

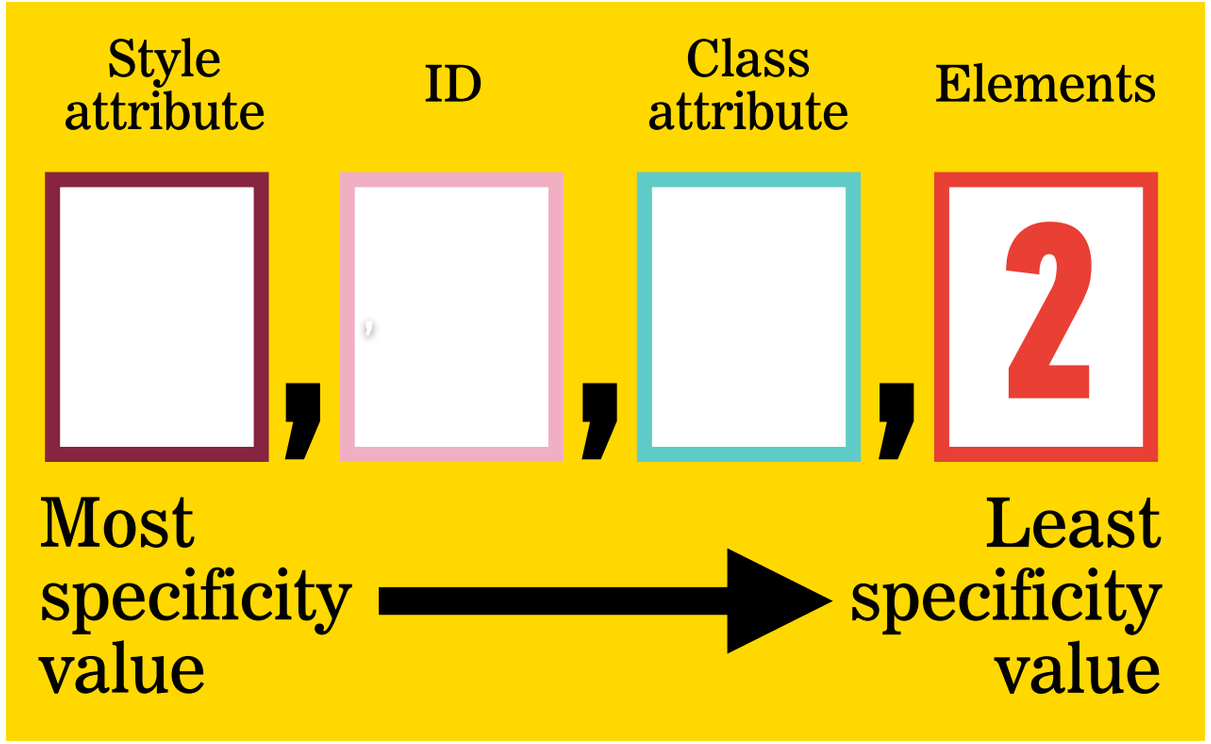
CSS:

```
li a {
  text-decoration: underline;
}

p a {
  text-decoration: none;
}
```

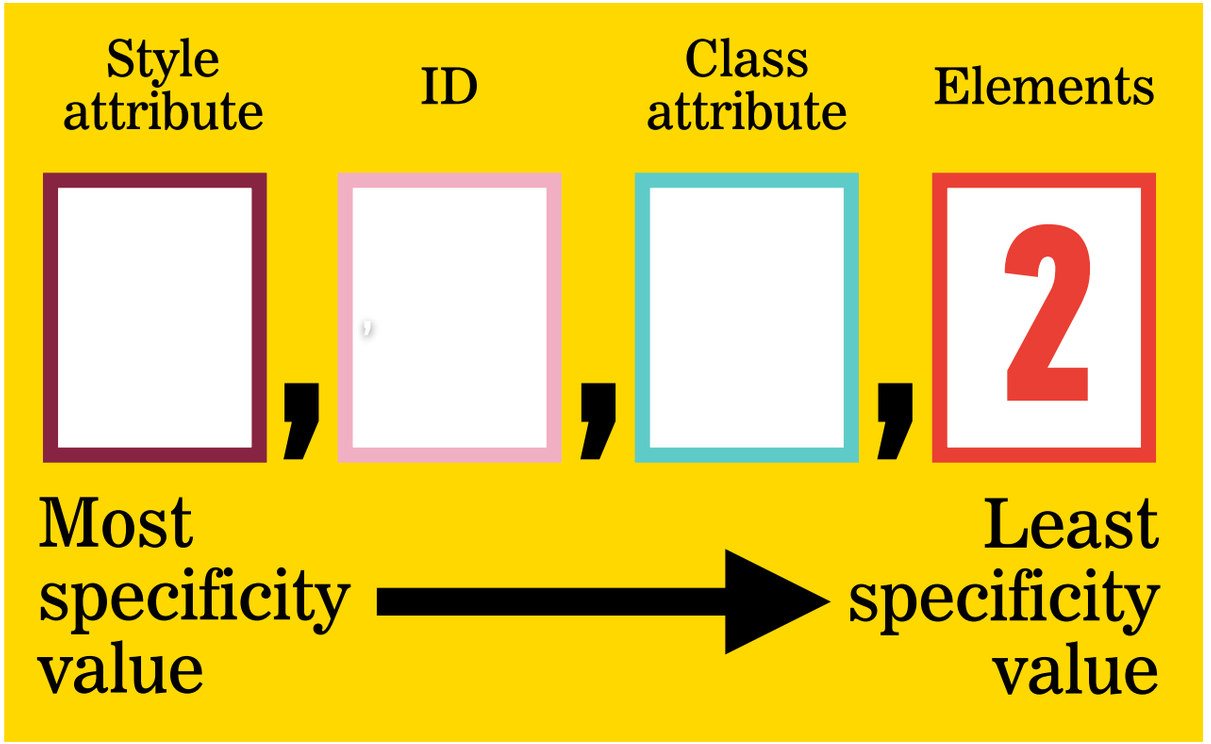
LAST RULE

TIE?!

li a



p a



CSS CASCADING — THE SPECIFICITY GAME

HTML:

```
<li>
  <p>Visit this <a href="">Cool Site</a></p>
</li>
```

CSS:

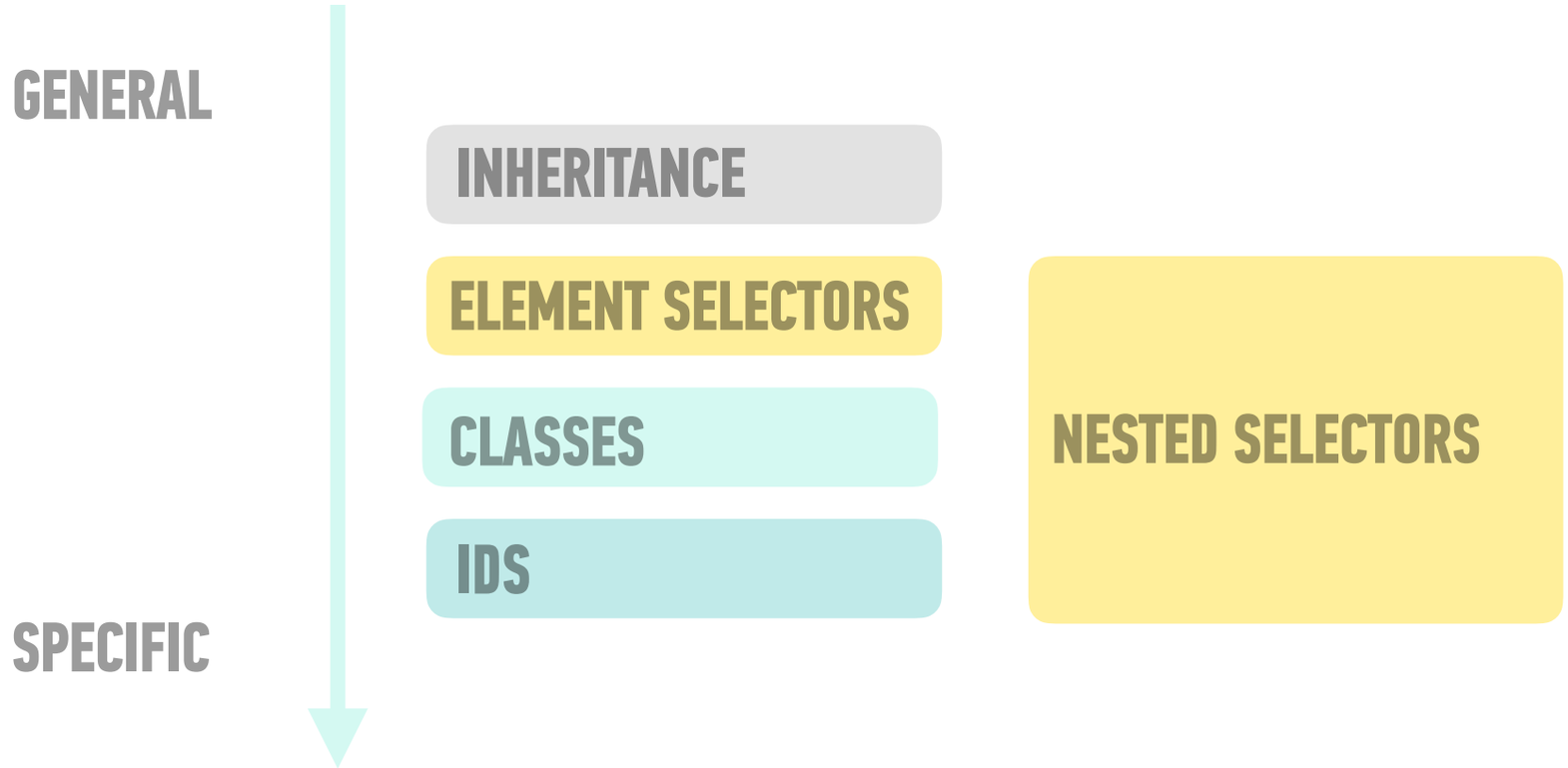
```
li a {
  text-decoration: underline;
}

p a {
  text-decoration: none;
}
```

CONFLICT!! — WHEN TWO RULES APPLY TO THE SAME ELEMENT

If the two selectors have the same importance and specificity, the one that comes later in the stylesheet will take precedence.

MORE ABOUT CASCADING — GENERAL TO SPECIFIC

- CSS rules cascade downward until they are canceled out by another rule.



- CSS rules are able to override one another and cancel each other out, depending on their order.

THINGS YOU SHOULD USE IF YOU WANT TO BE

INLINE STYLES

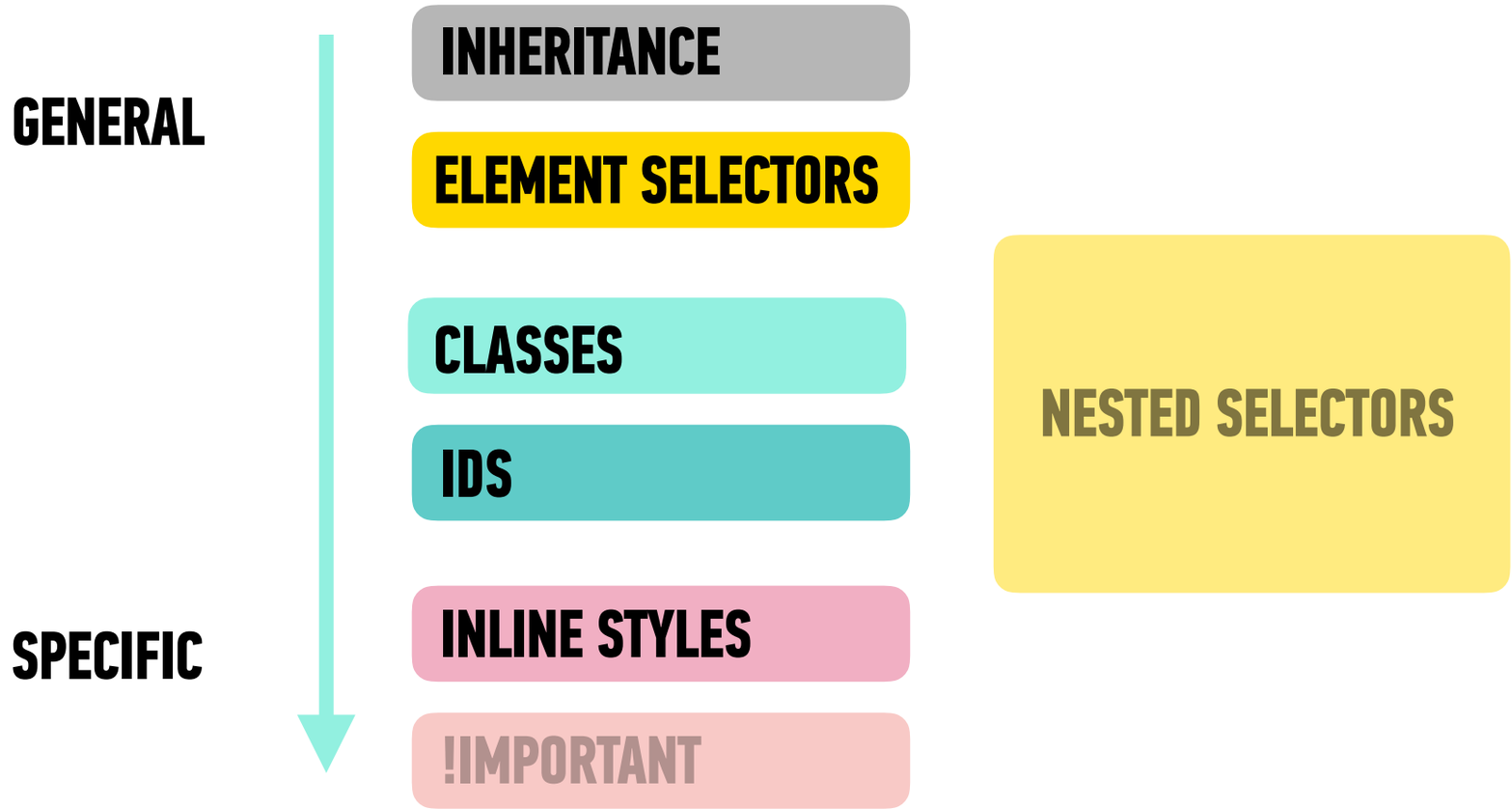
INLINE STYLES

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

BAD!!!!

MORE ABOUT CASCADING

- CSS rules cascade downward until they are canceled out by another rule.



- CSS rules are able to override one another and cancel each other out, depending on their order.

IMPORTANCE

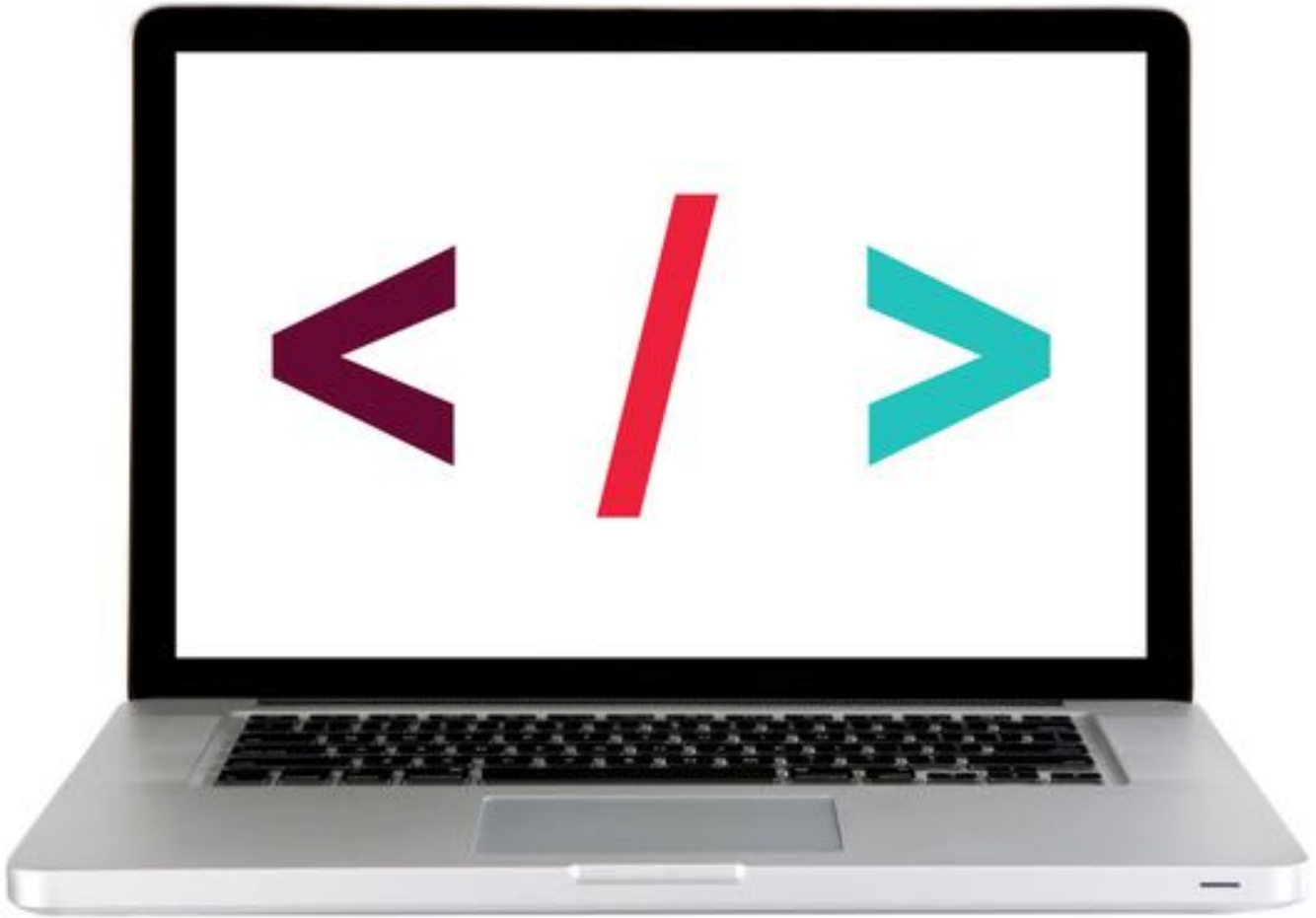
CSS IMPORTANCE

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

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

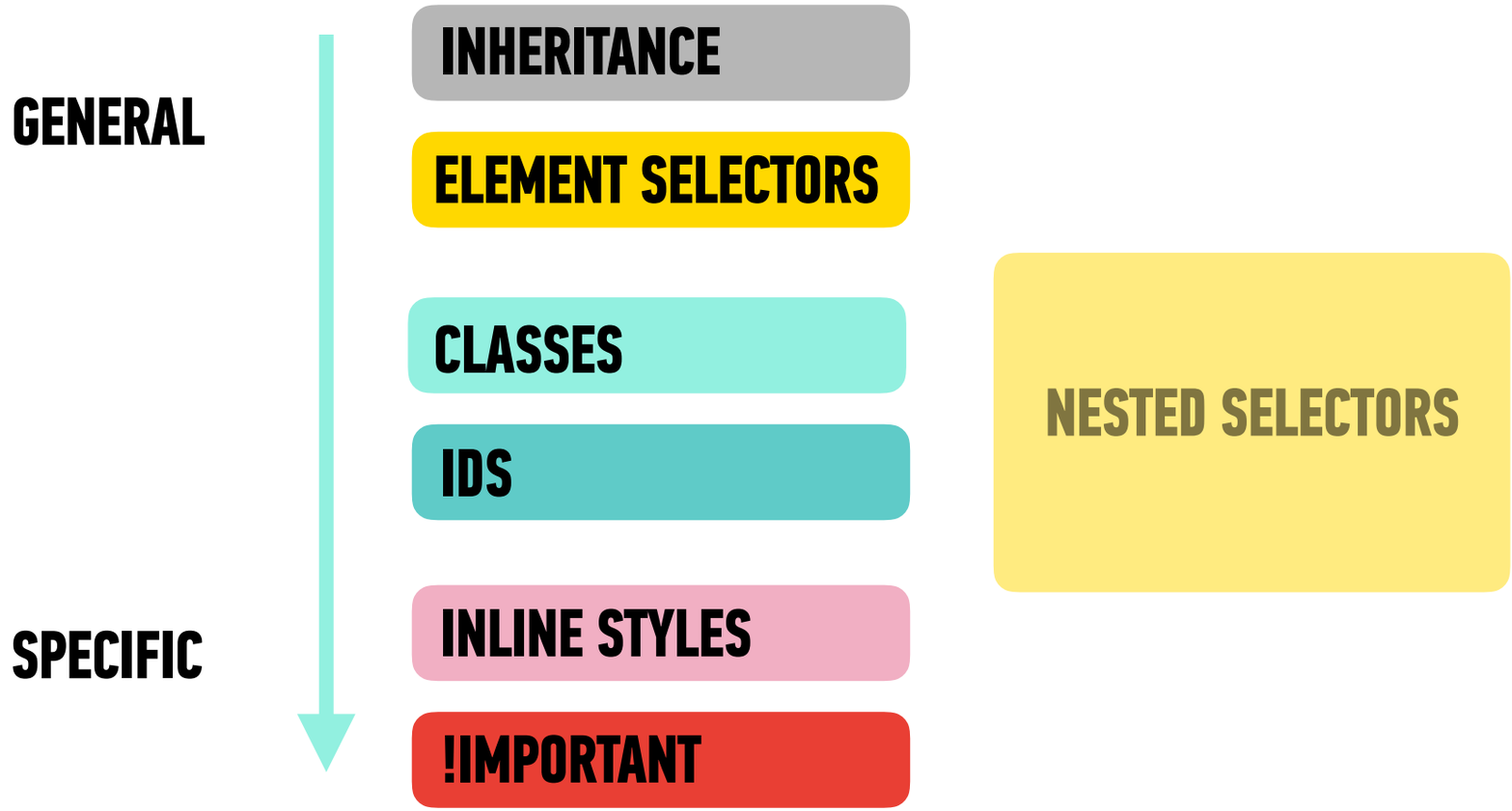
ONLY WHEN NECESSARY!!!
(Which is almost never)

LET'S TAKE A CLOSER LOOK



MORE ABOUT CASCADING

- CSS rules cascade downward until they are canceled out by another rule.



- CSS rules are able to override one another and cancel each other out, depending on their order.

CSS CASCADING — THE SPECIFICITY GAME

HTML:

```html
<ul id="about">
  <li style="color: red">Wednesday</li>
</ul>
```

CSS:

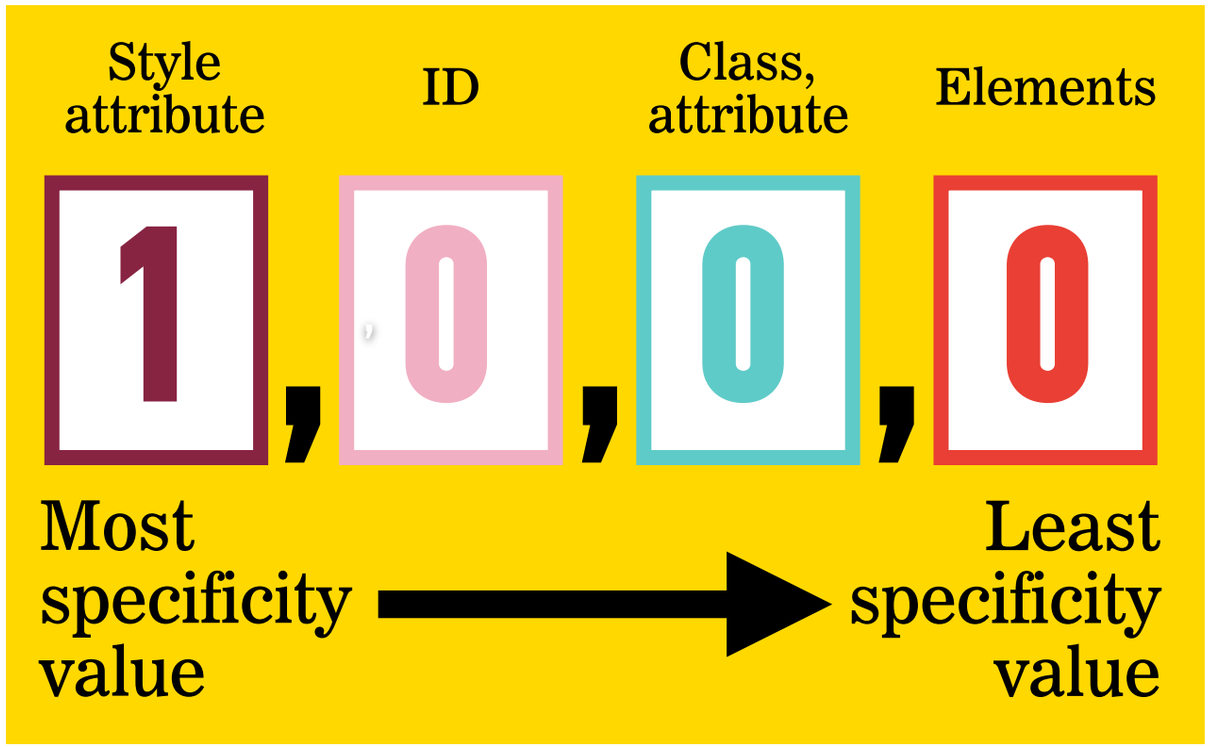
```css
#about li {
  color: pink;
}
```

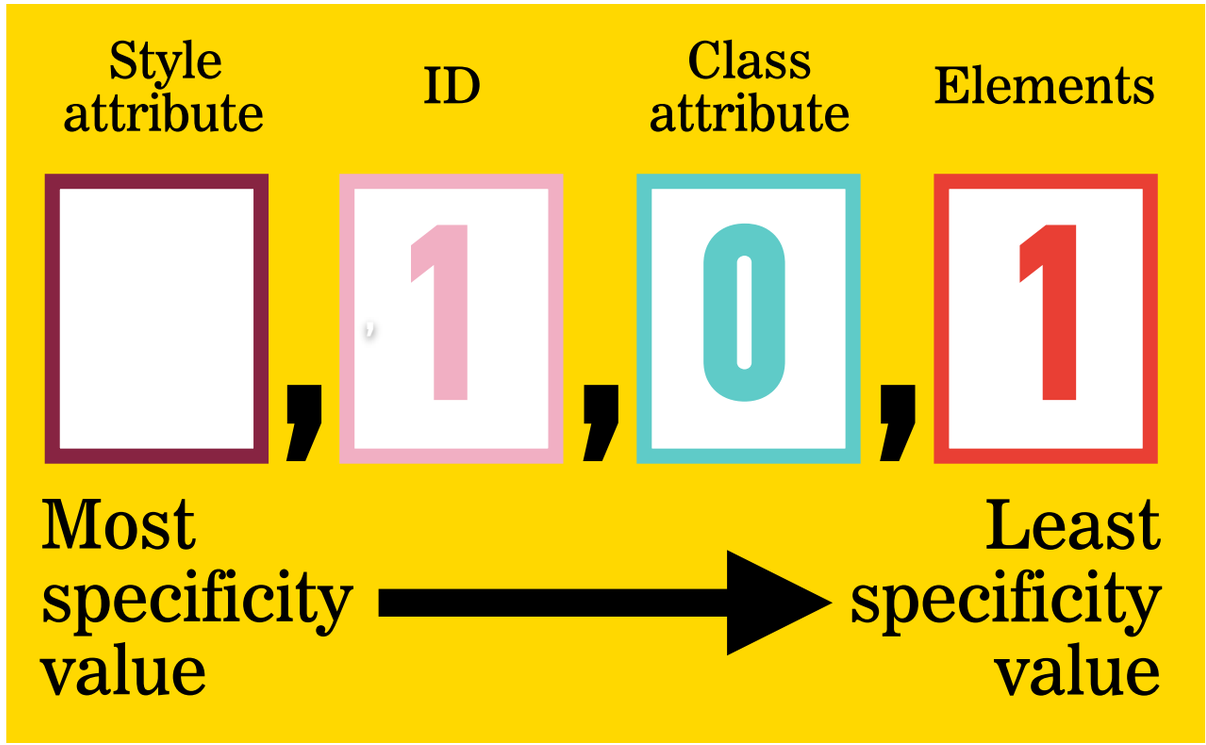
CSS CASCADING

WINNER!

`<li style="color: red"></li>`

`#about li`





CSS CASCADING — THE SPECIFICITY GAME

HTML:

```
<ul id="about">
  <li style="color: red">Wednesday</li> <!-- Score: 1000 -->
</ul>
```

CSS:

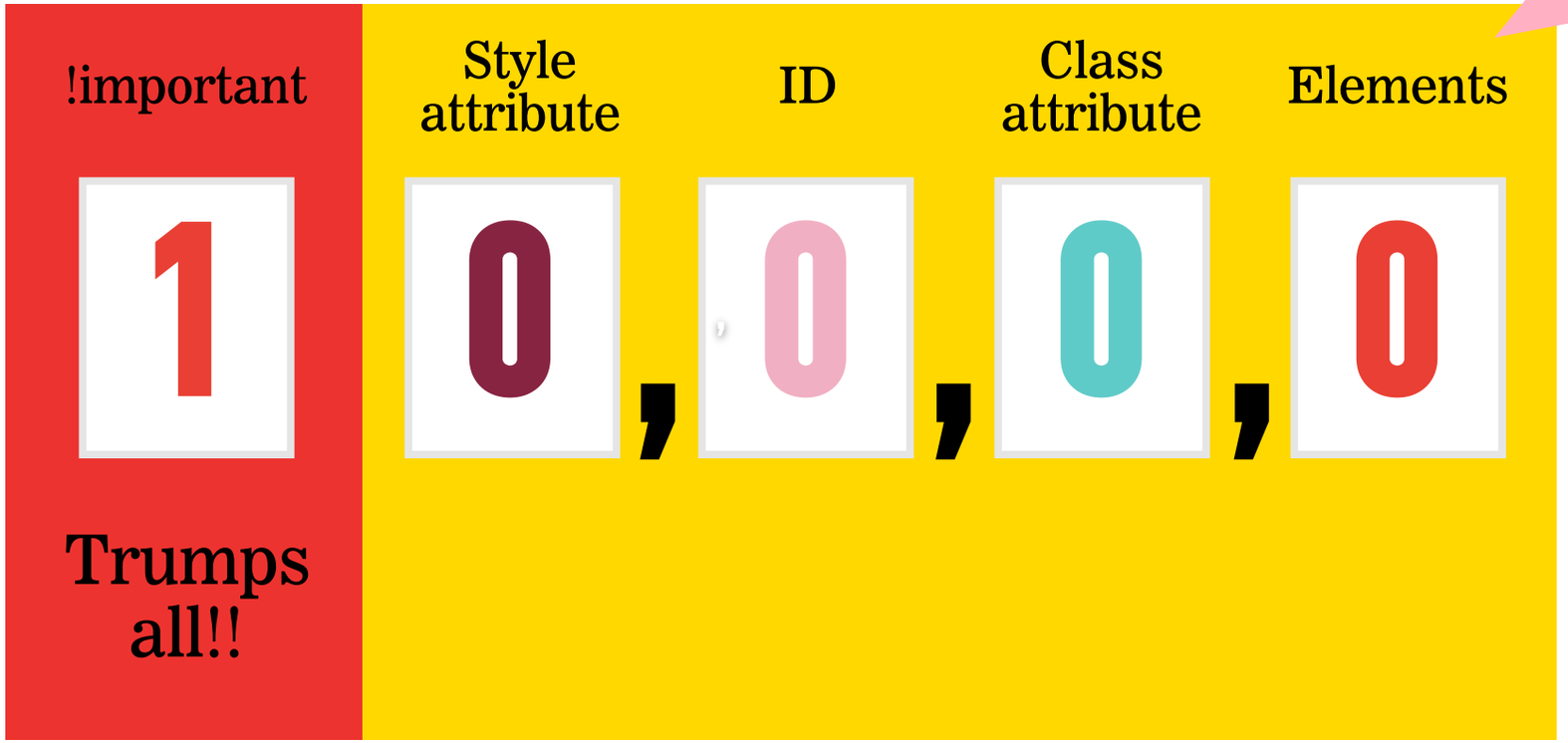
```
#about li {  /* Score: 101 */
  color: pink;
}
```

CSS CASCADING — THE SPECIFICITY GAME

!IMPORTANT

li {color: green *!important*;}

WINNER!



!important can only be overridden by another !important

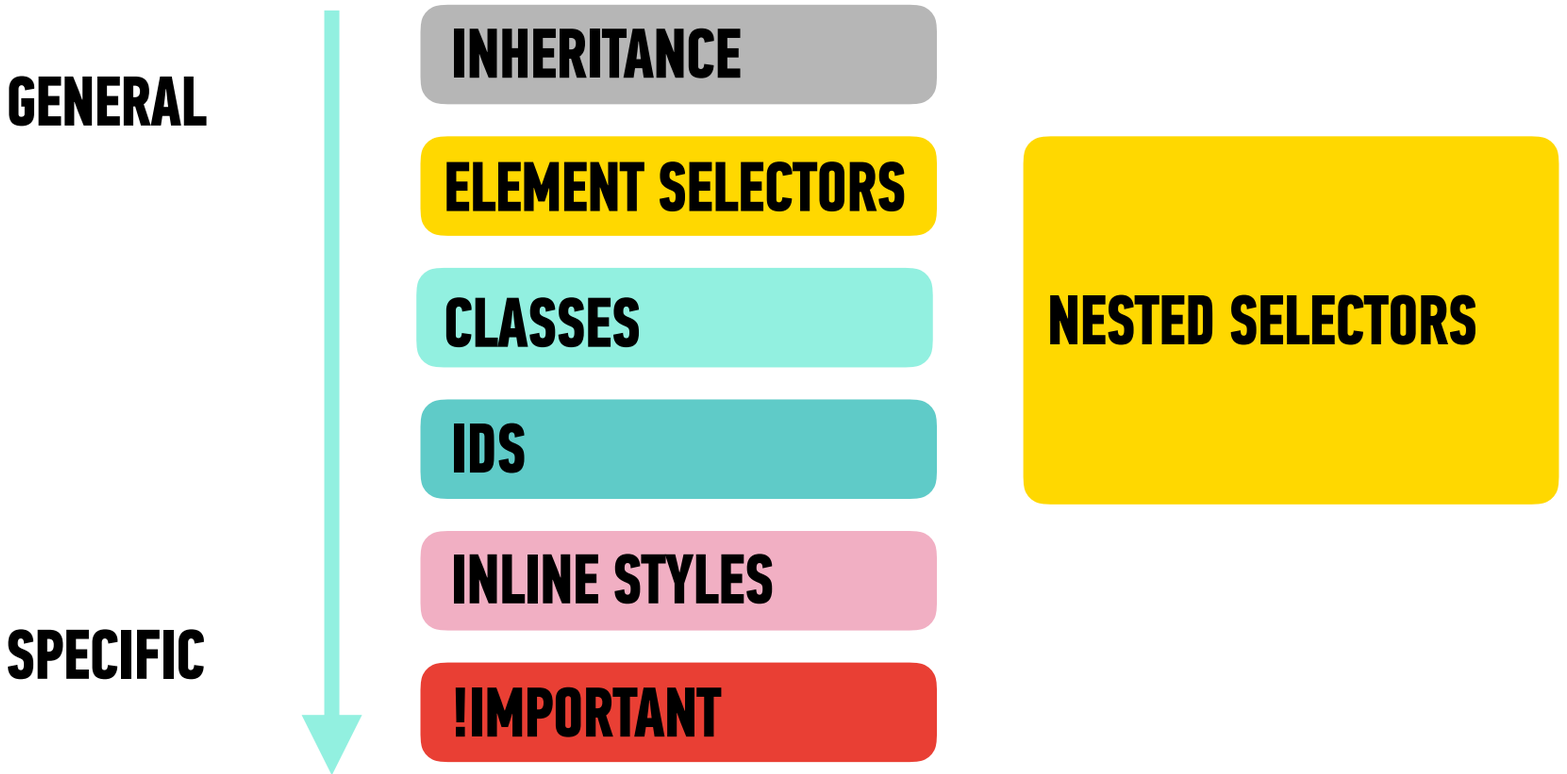
CSS CASCADING — THE SPECIFICITY GAME

```
li {
  color: green !important;
}
```

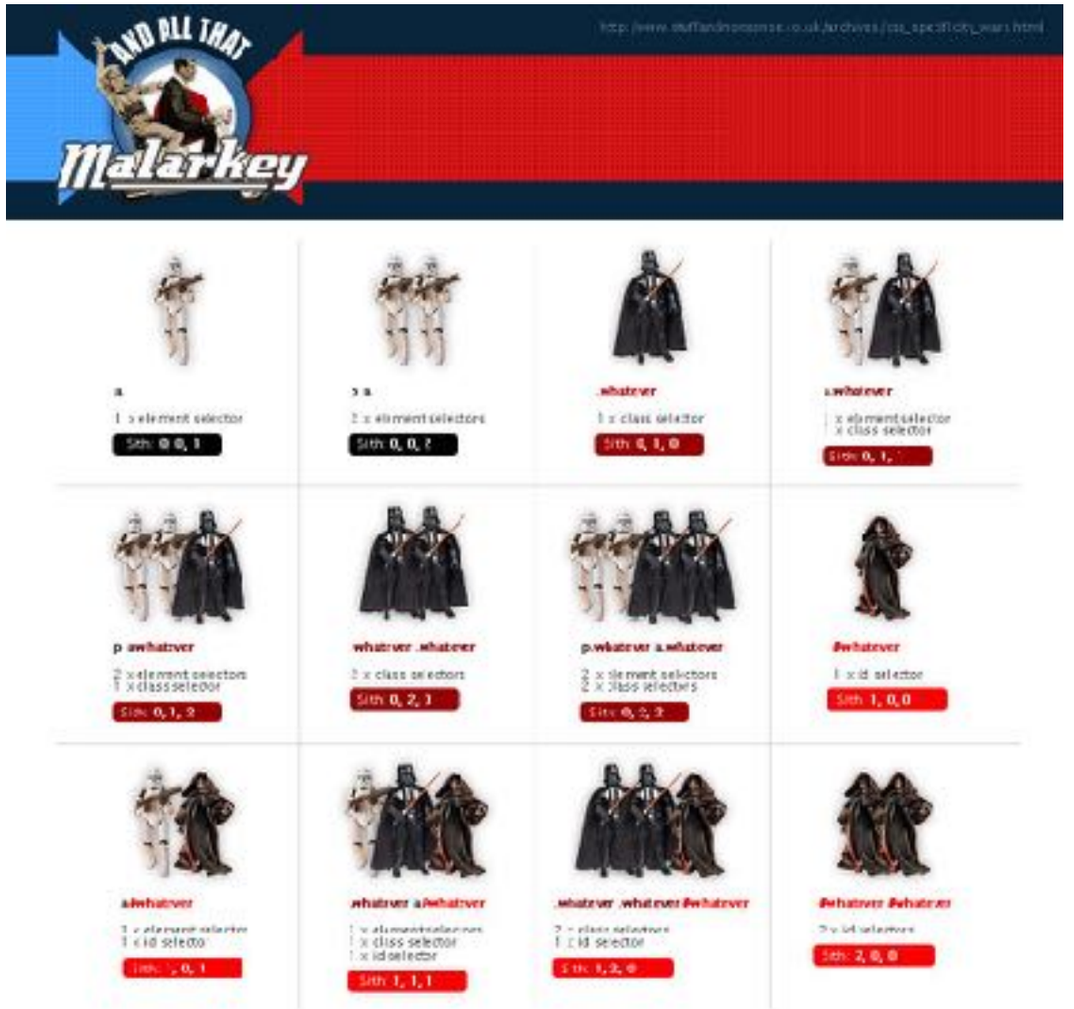
CSS CASCADING — THE SPECIFICITY GAME

SPECIFICITY:

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



CSS CASCADING — THE SPECIFICITY GAME



ACTIVITY

KEY OBJECTIVE

- Explain CSS "cascading"
- Summarize "specificity" — i.e. when multiple style rules apply to the same element, how can we figure out which one will be applied?
- Challenge question: How does the cascading nature of CSS allow us to write less CSS in the long run?

TYPE OF ACTIVITY

- Turn and Talk

TASKS

2 min	1. Turn and talk to a partner
2 min	2. A couple pairs will share with the class

USING CHROME DEV TOOLS

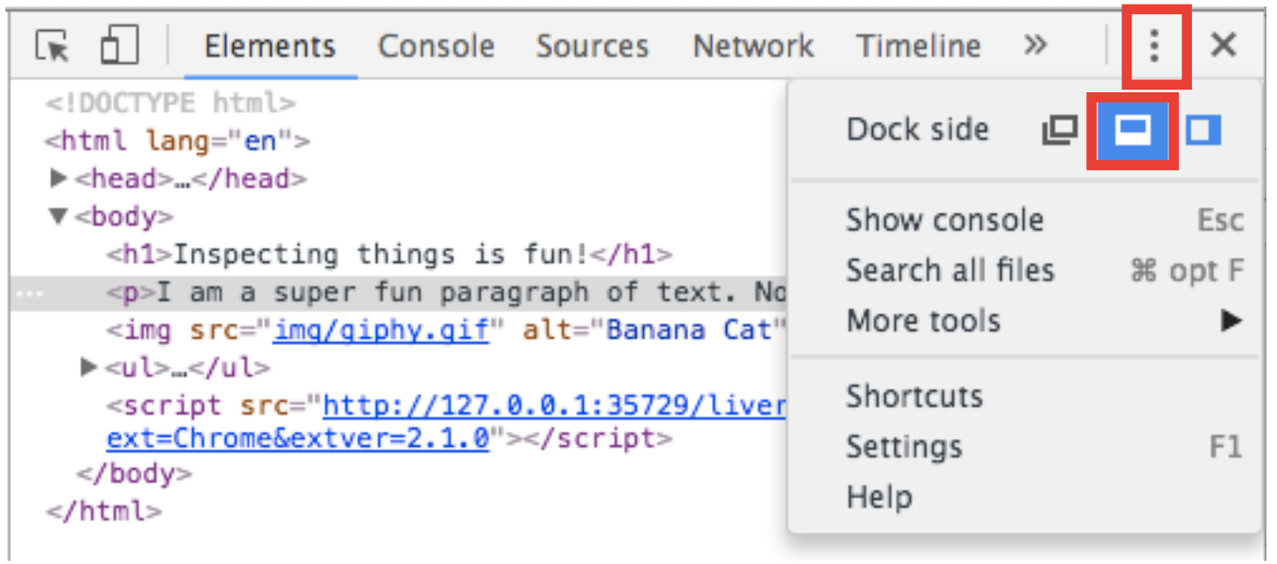
LET'S USE THE INSPECTOR!

There are several ways to open Chrome dev tools:

- Right click on an element and click "inspect"
- In Chrome, go to view > developer > Developer Tools
- Keyboard shortcut: Mac: *Cmd + Opt + I* Windows: *F12*, *Ctrl + Shift + I*

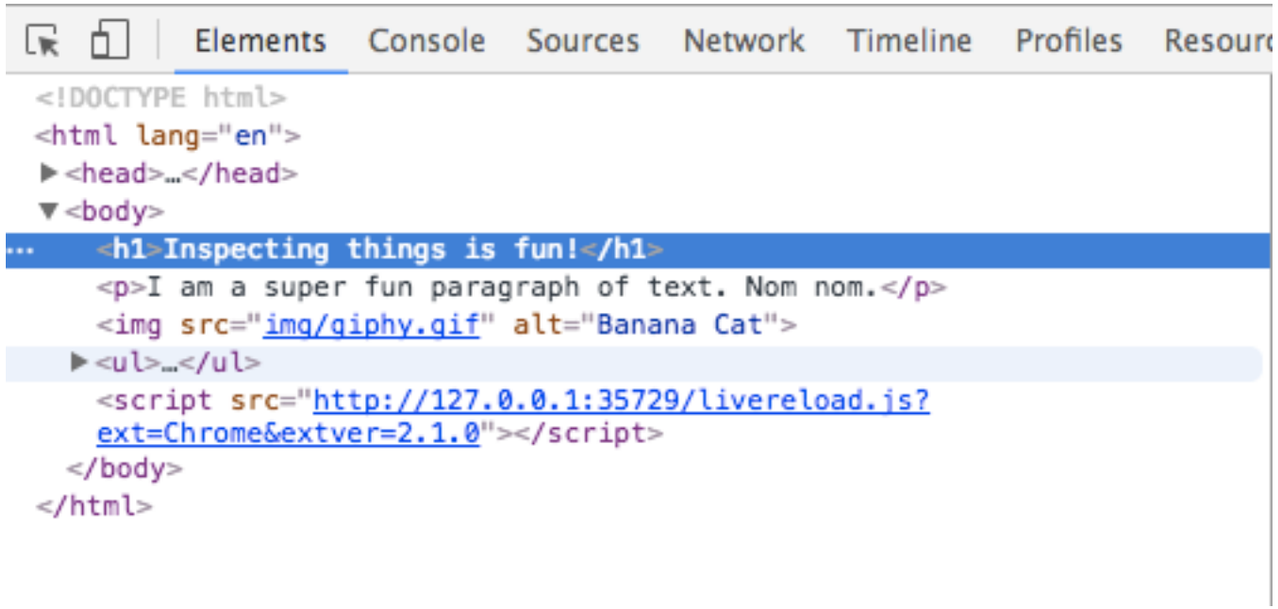
DOCK LOCATION

I find it easiest to move the dock to the bottom of the window.



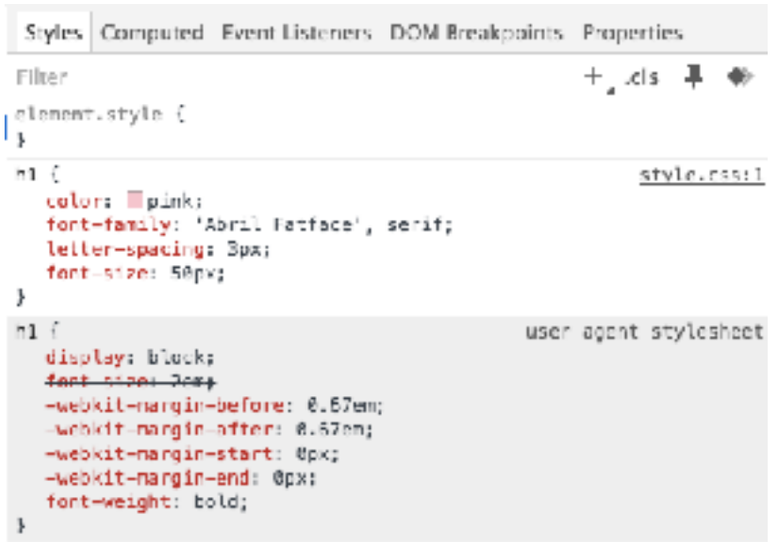
PICKING AN ELEMENT

Choose an element to inspect by clicking on the element in the "Elements" tab

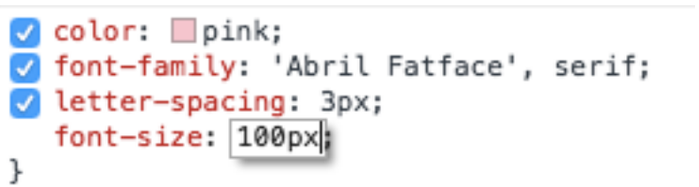


SEEING STYLES AND EXPERIMENTING

You can see what styles you've added (and default styles added by the browser) to the element in the "styles" panel.

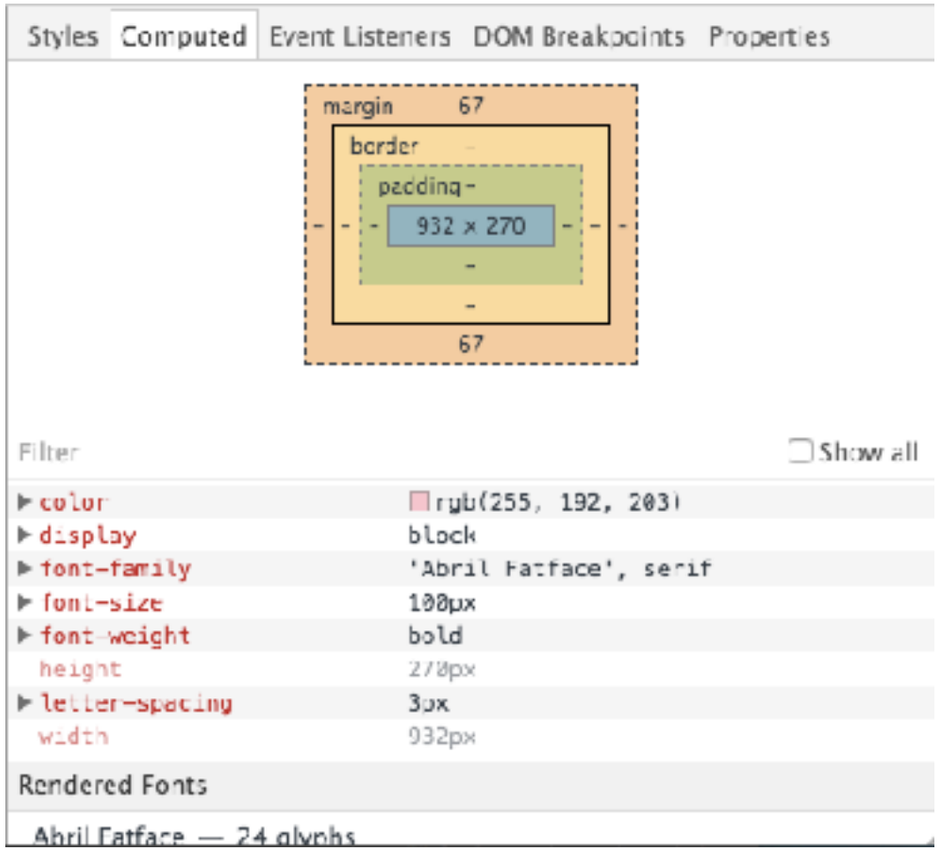


You can also experiment here!



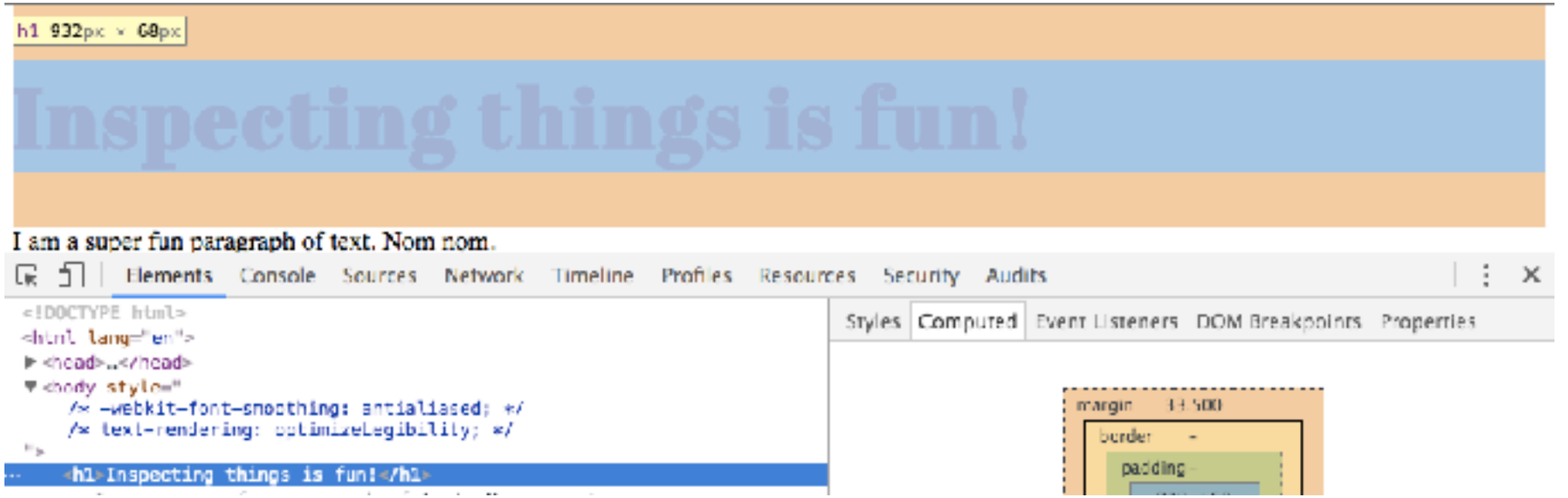
FINAL RENDERED STYLES AND DIMENSION

In the "computed" tab you can see all the styles that are being rendered on screen for an element. You can also see dimensions for an element including width, height, padding, margin and border.



VISUALIZING DIMENSIONS

One of the things I find most helpful when working through layout issues is to hover over different elements and see where everything is.



Orange: margin
Green: padding
Also notice width and height in a little yellow box by the element

LET'S USE THE INSPECTOR!

EXERCISE

LOCATION

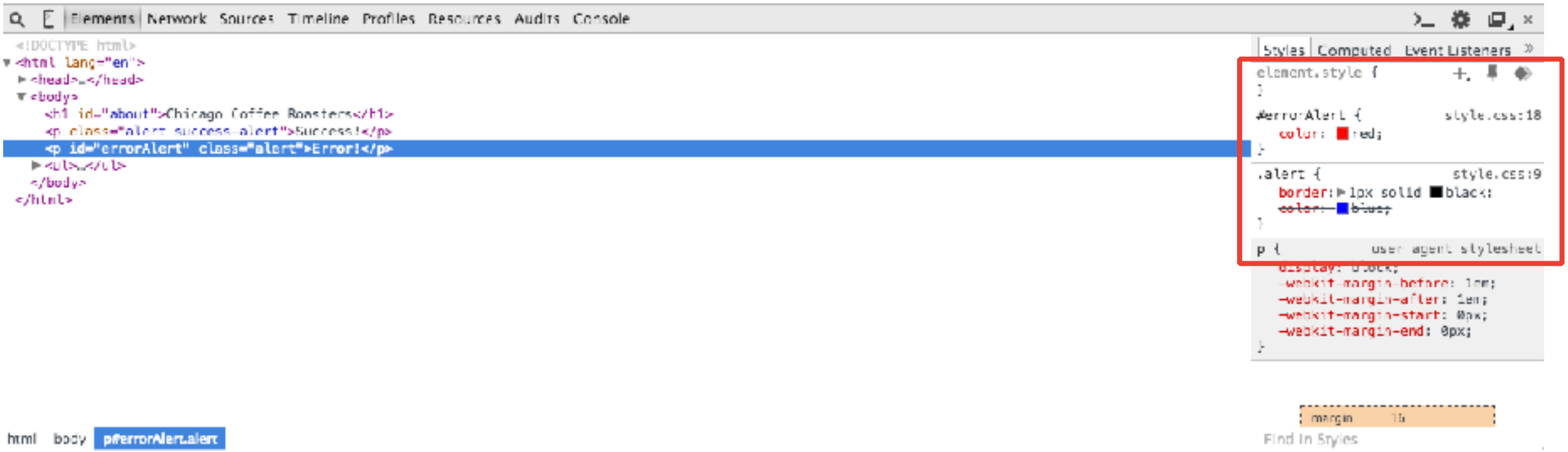
- starter_code_lesson_5 > inspect_element

LAB SESSION

8 min

1. What color is the h1? Can you find the rgb value?
2. What is the width and height of the h1? How much margin does it have on each side?
3. Hover over the h1 to visualize these dimensions.
4. What is the margin-top of the p element? Margin-bottom? Are these styles added in our stylesheet or default styles given by the browser (in the user agent stylesheet)
5. What is the background-color for each li? How much padding and margin does each li have?
6. Hover over an li to visualize these dimensions.
7. What is the width/height of the img?

INSPECT ELEMENT



To access the elements panel (in Chrome): Right click > Inspect Element

MONUMENT MAGAZINE

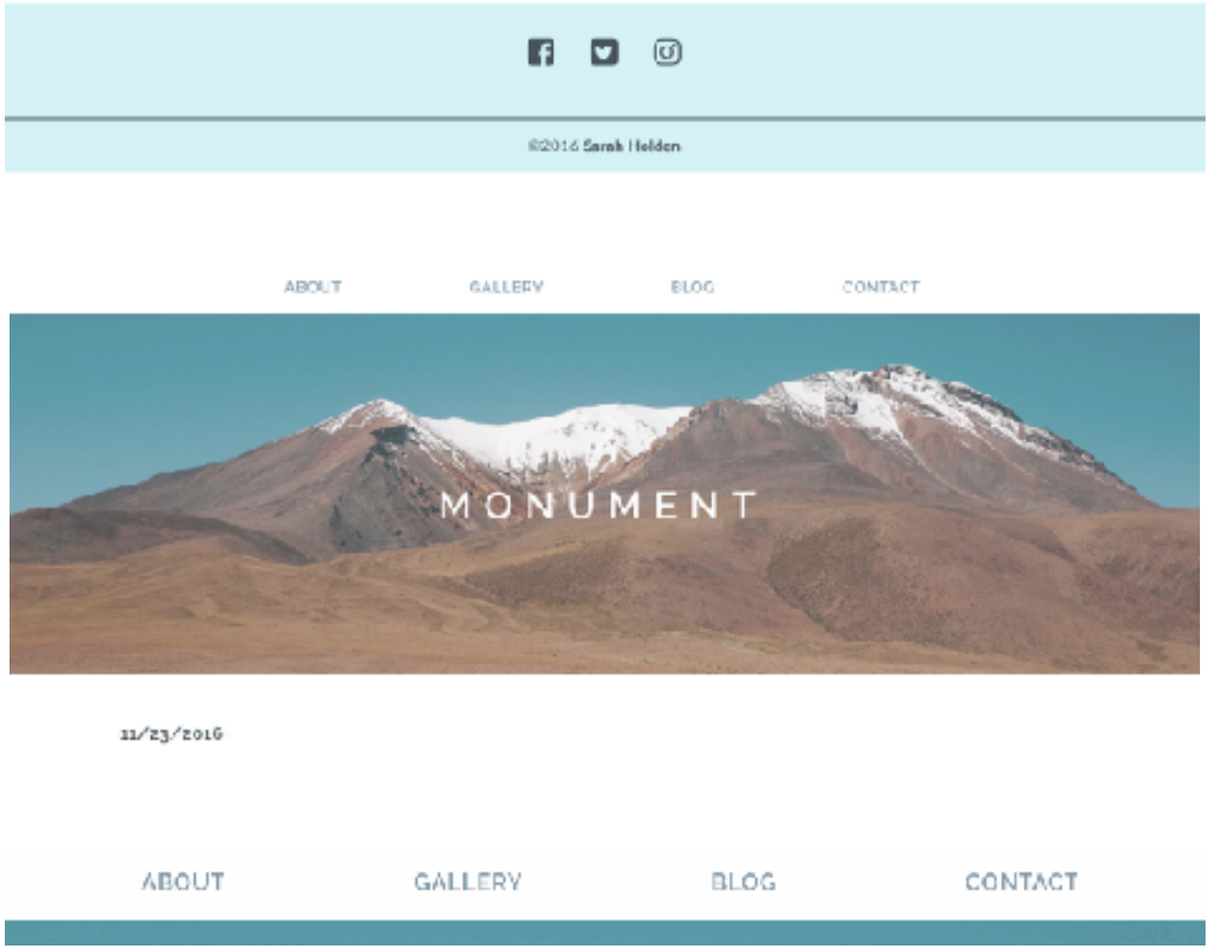
OVERVIEW

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.

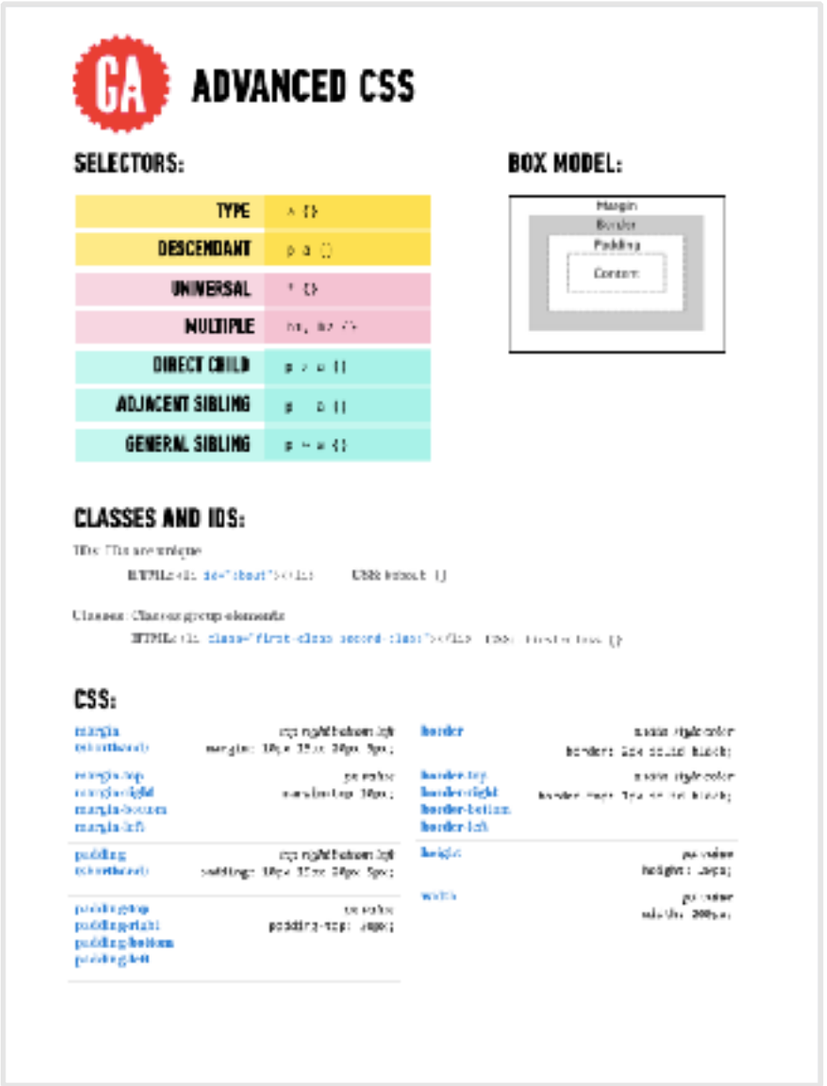
MONUMENT

BONUS TASKS:


©2016 Sarah Holden
ABOUT
GALLERY
BLOG
CONTACT
MONUMENT
11/23/2016
ABOUT
GALLERY
BLOG
CONTACT

LAB — MONUMENT MAGAZINE

ACTIVITY - DON'T FORGET YOUR HANDOUT!



GA ADVANCED CSS

SELECTORS:

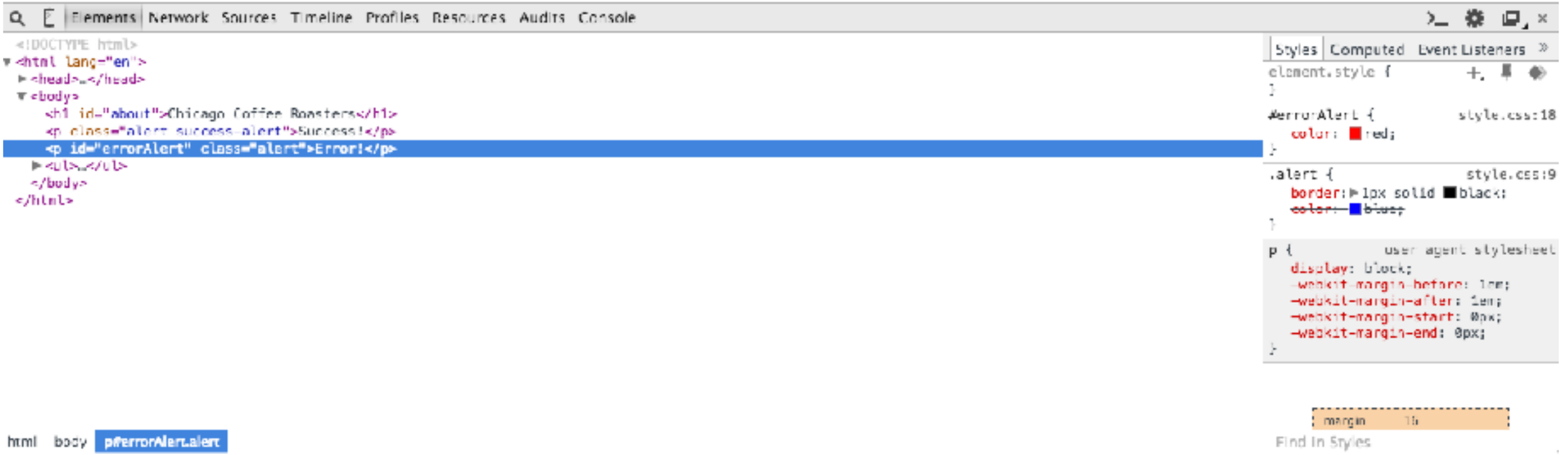
TYPE	
DESCENDANT	
UNIVERSAL	
MULTIPLE	
DIRECT CHILD	
ADJACENT SIBLING	
GENERAL SIBLING	

BOX MODEL:

CLASSES AND IDS:

CSS:

ACTIVITY - KEEP THE INSPECT ELEMENT PANEL OPEN AT ALL TIMES!



To access the elements panel (in Chrome): Right click > Inspect Element

ACTIVITY

EXERCISE

KEY OBJECTIVE

- Recreate the Monument Magazine site, using monument_blog.png as a reference

TASKS

1. Look through the provided markup and note new elements (article, aside, footer, nav). We'll be covering these next lesson, but look them up if you're wondering what they are used for.
2. Add a webfont (fonts/weights are listed in monument_blog.png)
3. Add styles! Start general and only get more specific when you need to. **You should not need to add any classes/ids in blog.html.**
4. **Google and implement a background-image in the header**
5. **Bonus 1:** Google and implement social icons in the footer using FontAwesome (image supplied)
6. **Bonus 2:** 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 1	HTML Basics / CSS Basics
WEEK 2	CSS Selectors / Layout
WEEK 3	Lab / Refactor & Lab

NEXT CLASS:

EXIT TICKETS!