

FEWD – LESSON 2

BEFORE WE GET STARTED:

- Register for a GitHub account and download desktop app – if you haven't already
- Download and install Sublime Text and Chrome – if you haven't already
- Clone or download the lesson 2 repository

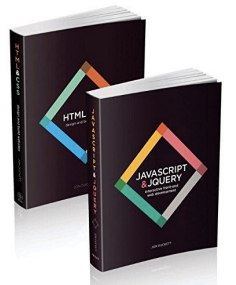
https://github.com/Dt1431/FEWD_Lesson_02

 Clone in Desktop

 Download ZIP

RECOMMENDED COURSE MATERIALS:

- [Jon Duckett - Web Design with HTML, CSS, JavaScript and jQuery Set](#)



FEWD – LESSON 2

CSS BASICS

Dave Turner, Co-founder @ Waitbot

AGENDA



- More HTML Basics
- Intro to CSS
- LAB: About me landing page
- Wrap Up

LEARNING OBJECTIVES

- Describe the DOM and draw simple DOM tree.
- Predict image paths and apply relative paths to and <a> tags.
- Differentiate between the various image file types
- Apply and explain CSS “cascade” including: importance, specificity and inheritance.
- Differentiate between basic web color principles: RGB, RGBA, hexadecimal color, HSL.

FEWD – LESSON 2

MORE HTML BASICS

TIPS FOR HTML SUCCESS

- Create your opening and closing tags together
- Indent nested elements, i.e. tags within tags
- Beware of the `
` tag
- Friends don't let friends leave text unwrapped
- Keep nerding out!



NAVIGATION

- Used to contain a set of navigational links.
- Not all groups of links on a page need to be in a nav element — only sections that consist of major navigational blocks

`<nav>`

HTML ENTITY CODES

Use HTML [entity codes](#) to add special characters to your HTML.

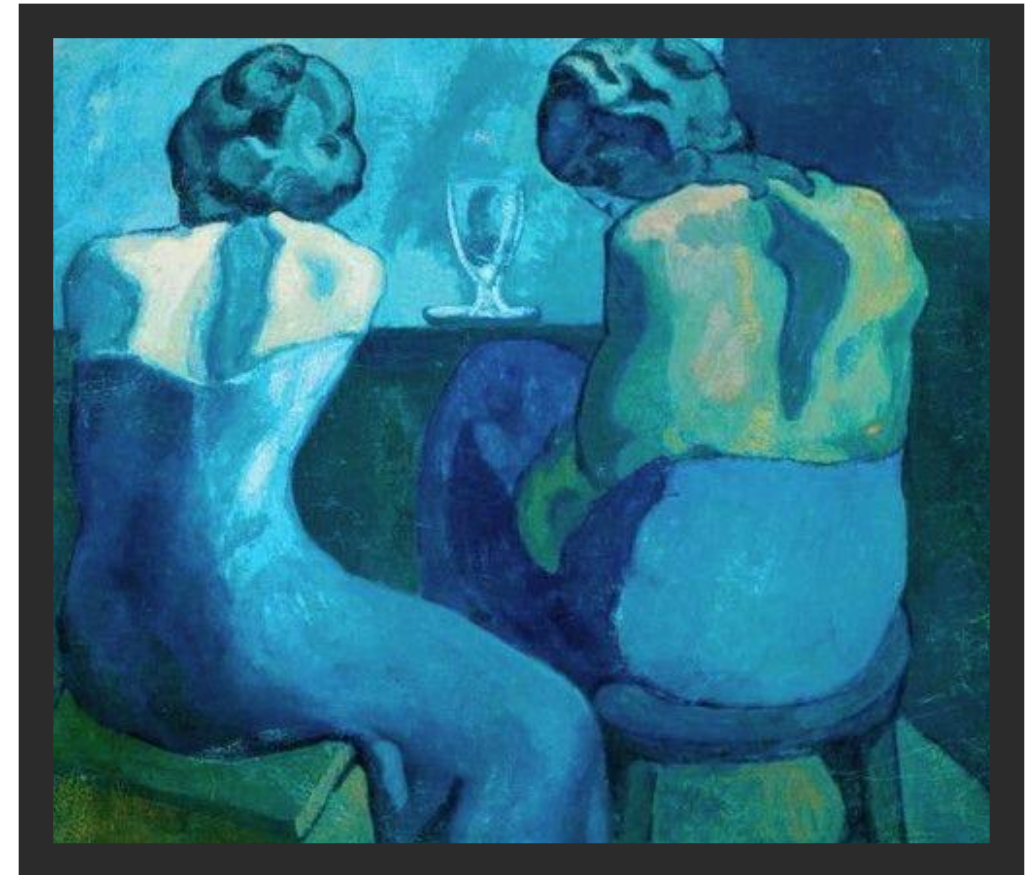
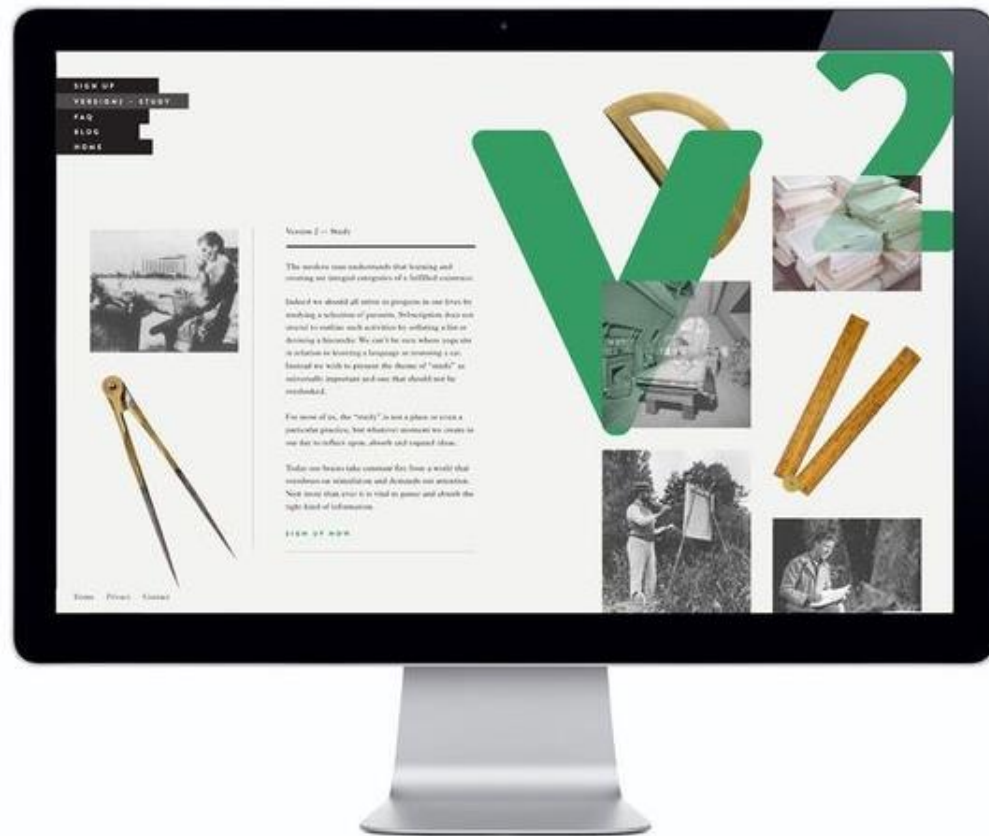
<code>&Tab;</code>	<code>&NewLine;</code>	! <code>&excl;</code>	" <code>&quot; &QUOT;</code>	# <code>&num;</code>	\$ <code>&dollar;</code>	% <code>&percent;</code>	& <code>&amp; &AMP;</code>
' <code>&apos;</code>	(<code>&lpar;</code>) <code>&rpar;</code>	* <code>&ast; &midast;</code>	+ <code>&plus;</code>	, <code>&comma;</code>	. <code>&period;</code>	/ <code>&sol;</code> <code>&#x0002F;</code> <code>&#47;</code>
:	; <code>&semi;</code>	< <code>&lt; &LT;</code>	= <code>&equals;</code>	> <code>&gt; &GT;</code>	? <code>&quest;</code>	@ <code>&commat;</code>	[<code>&lqb; &lbrack;</code>
\ <code>&bsol;</code>] <code>&rsqb; &rbrack;</code>	^ <code>&Hat;</code>	— <code>&lowbar;</code>	` <code>&grave;</code> <code>&DiacriticalGrave;</code>	{ <code>&lcurb; &lbrace;</code>	 <code>&verbar; &vert;</code> <code>&VerticalLine;</code>	} <code>&rcub; &rbrace;</code>

LEARNING OBJECTIVES

- Describe the DOM and draw simple DOM tree.
- Predict image paths and apply relative paths to and <a> tags.
- Differentiate between the various image file types
- Apply and explain CSS “cascade” including: importance, specificity and inheritance.
- Differentiate between basic web color principles: RGB, RGBA, hexadecimal color, HSL.

WHAT IS THE DOM?

- People often think of web pages as though they are paintings or pages in a book, consisting of static text and images.



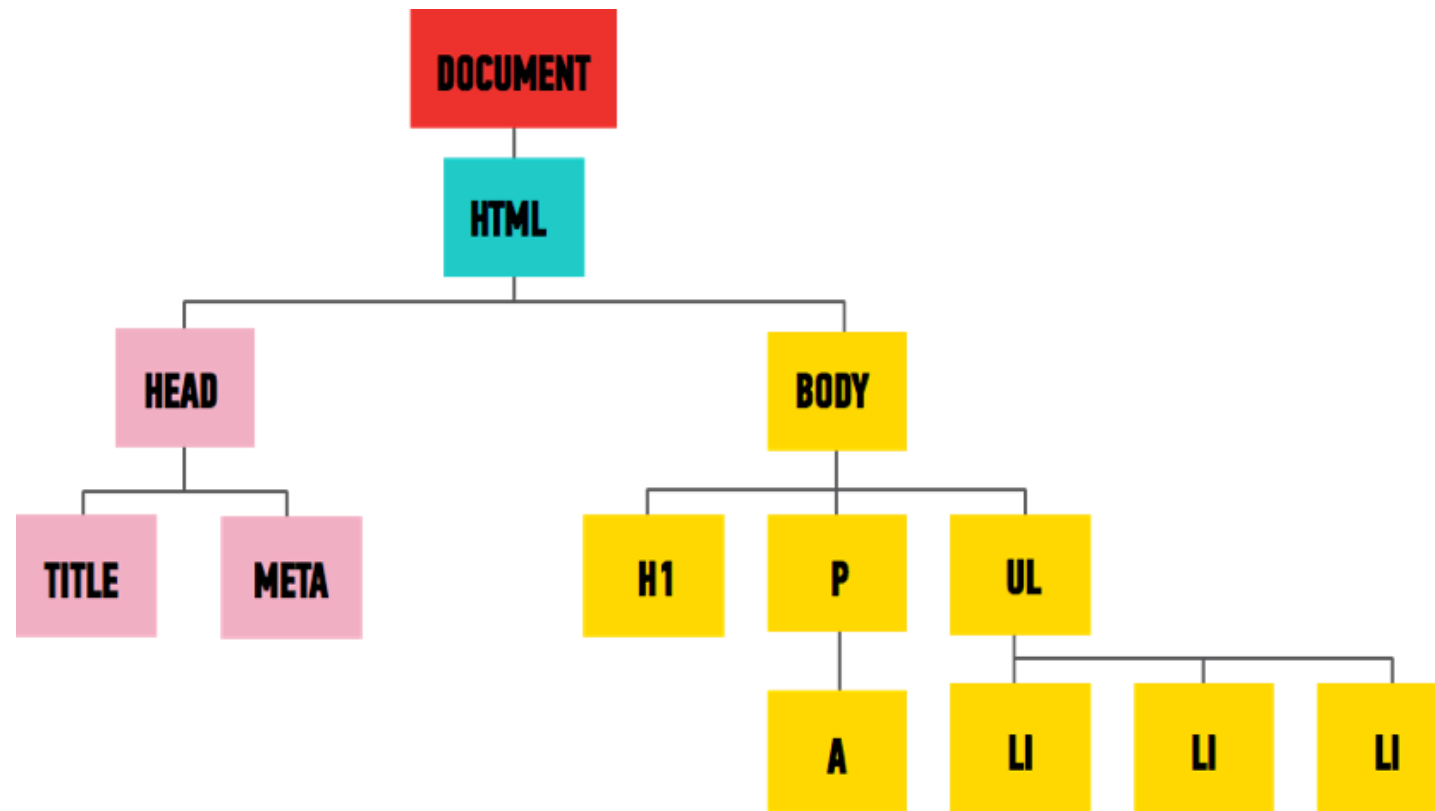
WHAT IS THE DOM?

- In fact web pages aren't static "paintings," but are built up of many parts.

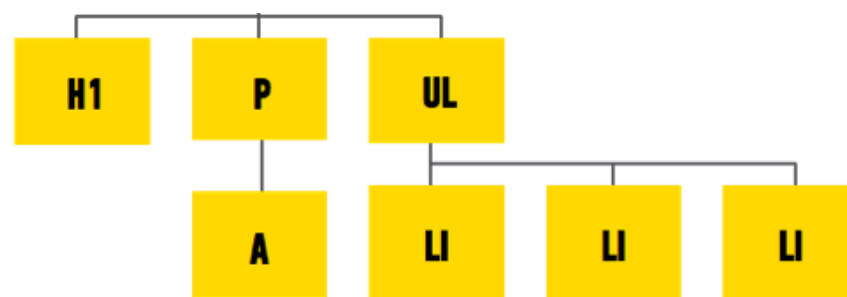


DOM TREE

- The Document Object Model (DOM) is the model that describes how all elements in an HTML page, like input fields, images, paragraphs etc., are related to the topmost structure: the document itself.

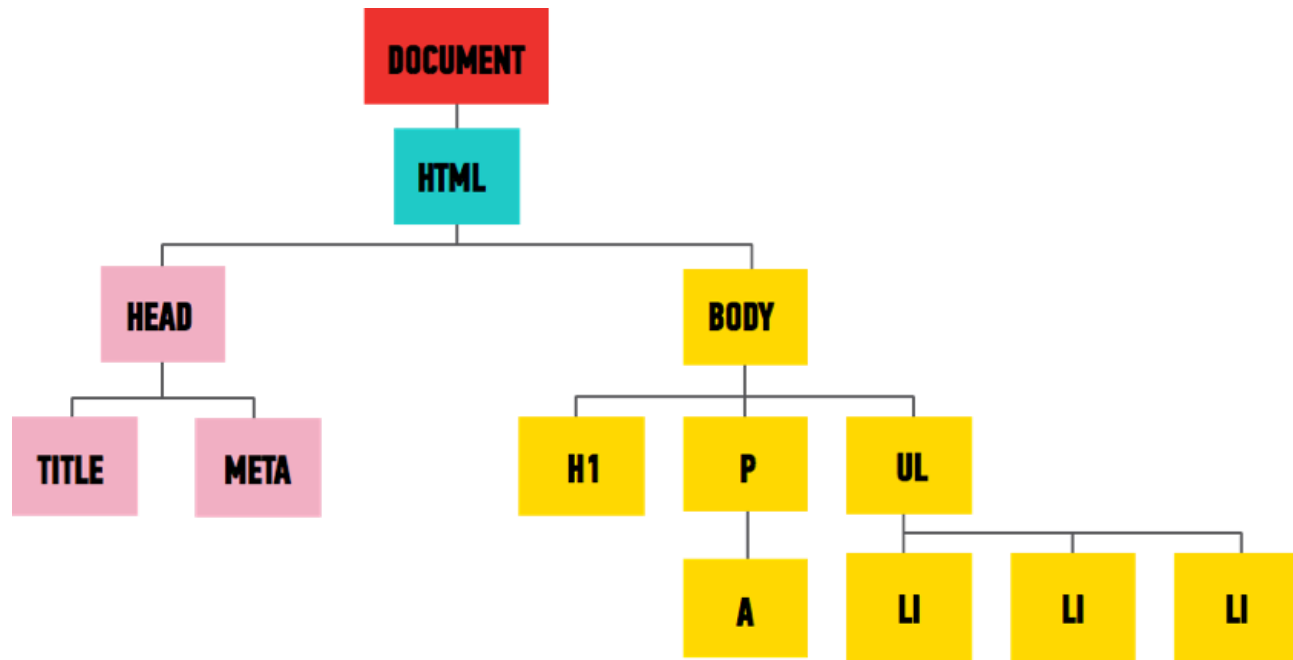


DOM TREE



```
<h1>Site title</h1>
<p>Bacon ipsum dolor amet brisket tail frankfurter cupim pig salami.
Biltong beef ribs ribeye short loin flank corned beef tri-tip beef ball tip
sausage jowl swine. Fatback porchetta strip steak doner chicken <a href="
http://www.jamieoliver.com/recipes/pork-recipes/pork-belly-roast/">pork
belly</a>. Rump picanha jowl ground round, pancetta doner boudin landjaeger
prosciutto meatloaf.</p>
<ul>
  <li>Bacon</li>
  <li>Chicken</li>
  <li>Meatloaf</li>
</ul>
```

HTML STRUCTURE



```
index.html
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8">
5     <title>Document</title>
6   </head>
7   <body>
8
9     <h1>Site title</h1>
10    <p>Bacon ipsum dolor amet brisket tail
    frankfurter cupim pig salami. Fatback
    porchetta strip steak doner chicken <a href="
    http://www.jamieoliver.com/recipes/pork-recipes
    /pork-belly-roast/">pork belly</a></p>
11    <ul>
12      <li>Bacon</li>
13      <li>Chicken</li>
14      <li>Meatloaf</li>
15    </ul>
16
17  </body>
18 </html>
```

LEARNING OBJECTIVES

- Describe the DOM and draw simple DOM tree.
- Predict image paths and apply relative paths to `` and `<a>` tags.
- Differentiate between the various image file types
- Apply and explain CSS “cascade” including: importance, specificity and inheritance.
- Differentiate between basic web color principles: RGB, RGBA, hexadecimal color, HSL.

IMAGES

- Images are added to the page using the element.
- This is a void element — it does not need a closing tag
- Two **required** attributes — src and alt

```

```

The forward slash before the closing ">" is optional for void elements in HTML5. It's considered best practice, though, to be consistent with your code — either always use a forward slash or never use a forward slash.



IMAGES

- The **src** attribute tells the browser where it can find an image.

Path to the image file



```

```

IMAGES

- The **alt** attribute provides a text description of the image which describes the image if you can't see it.
- This is also used by screenreaders and is known as the alt text

Text description in case you
can't see image



```

```

TYPES OF URLS

- There are two main types of URLs:



ABSOLUTE



RELATIVE

LINKING TO OTHER SITES - ABSOLUTE URLS

- When you link to a different website, the value of the href attribute will be the full web address for the site
- This is known as the absolute URL.



ABSOLUTE

Absolute URL

`Amazon`

``

LINKING TO OTHER PAGES ON THE SAME SITE — RELATIVE URLS

- When you are linking to other pages within the same site, you don't need to specify a domain name in the URL.
- You can use a shorthand known as the relative URL
- The relative URL indicates where pages are in relation to the current page

Relative URL

`About Page`

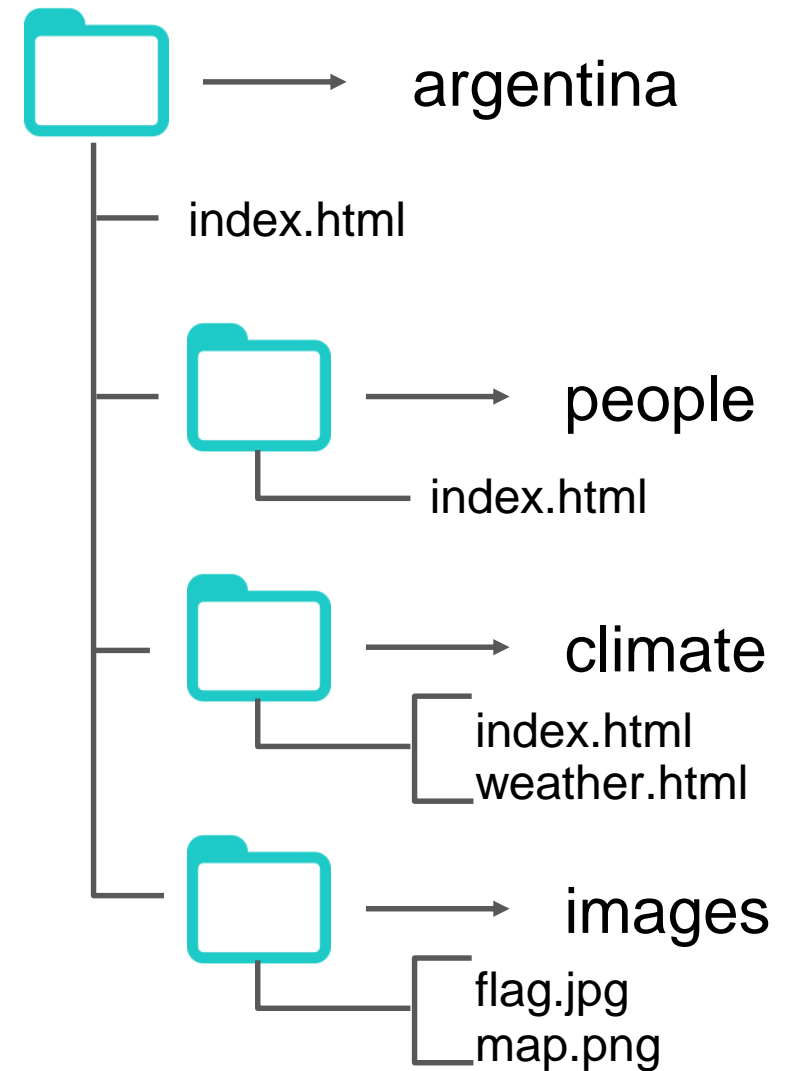
``



RELATIVE

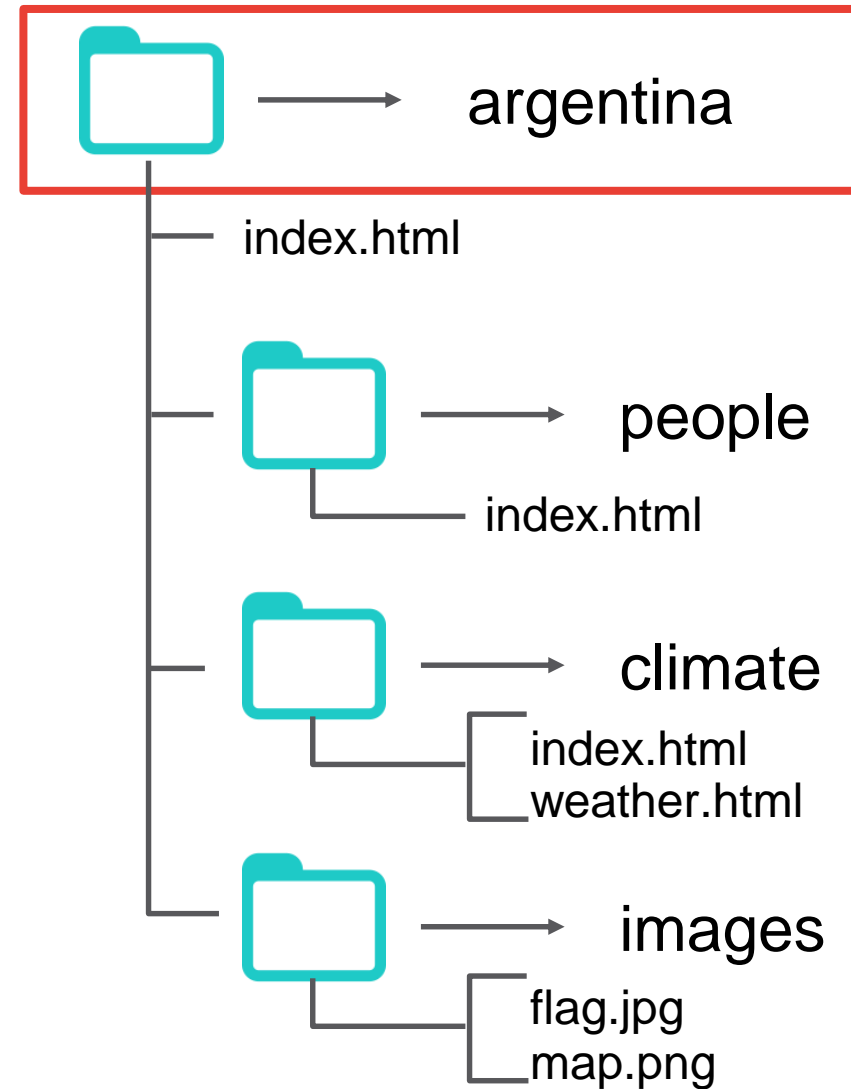
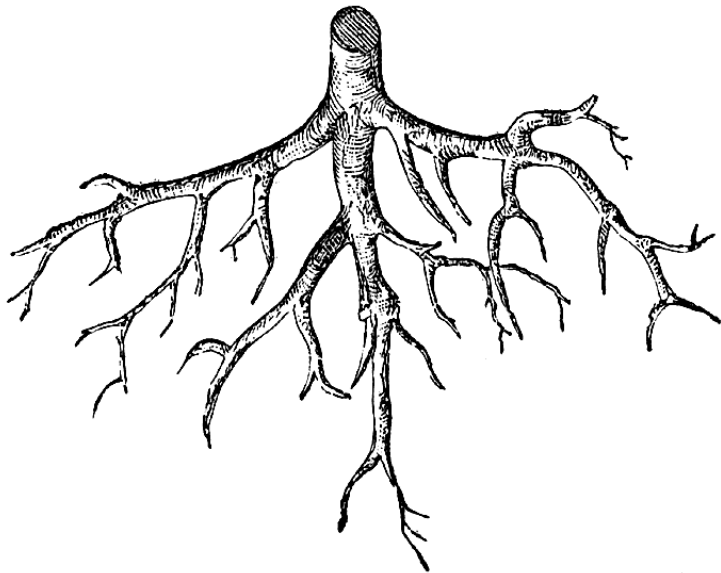
DIRECTORY STRUCTURE

- 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.



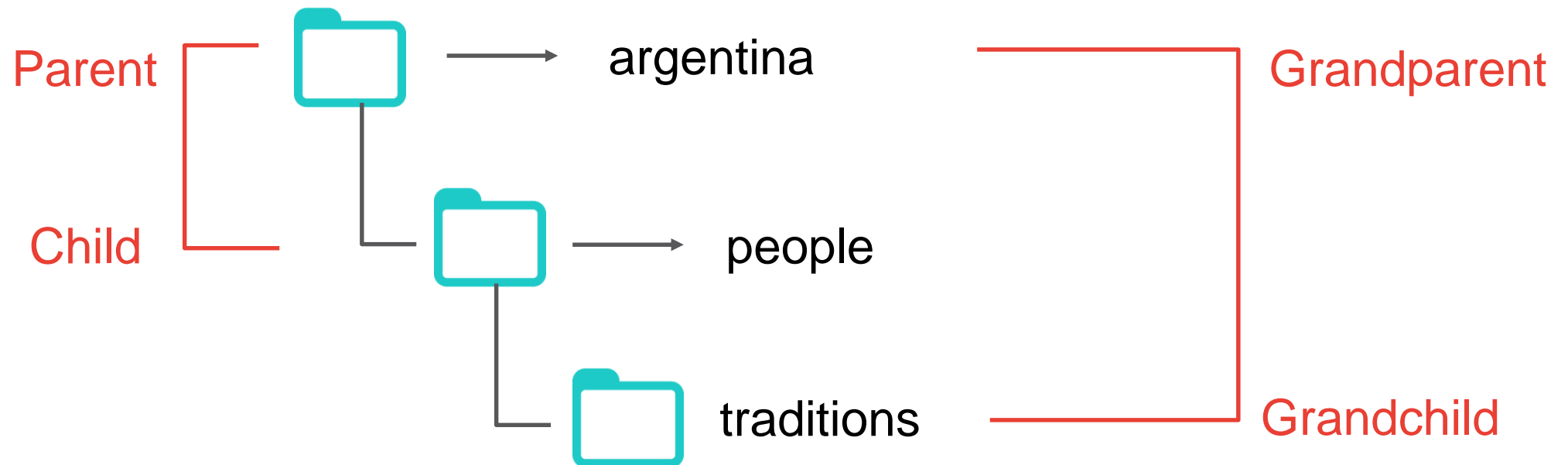
DIRECTORY STRUCTURE — ROOT

- The topmost folder is called the root folder. It contains all other files and folders for a website.



DIRECTORY STRUCTURE

- Relationships between files and folders can be described using similar language to that of a family tree — parent, child, grandparent, grandchild, etc.



LINKING TO OTHER PAGES ON THE SAME SITE — RELATIVE URLS

- Every page and every image on a site has a URL (Uniform Resource Locator).

```

```

RELATIVE URLS

FOLDER FILE IS IN:

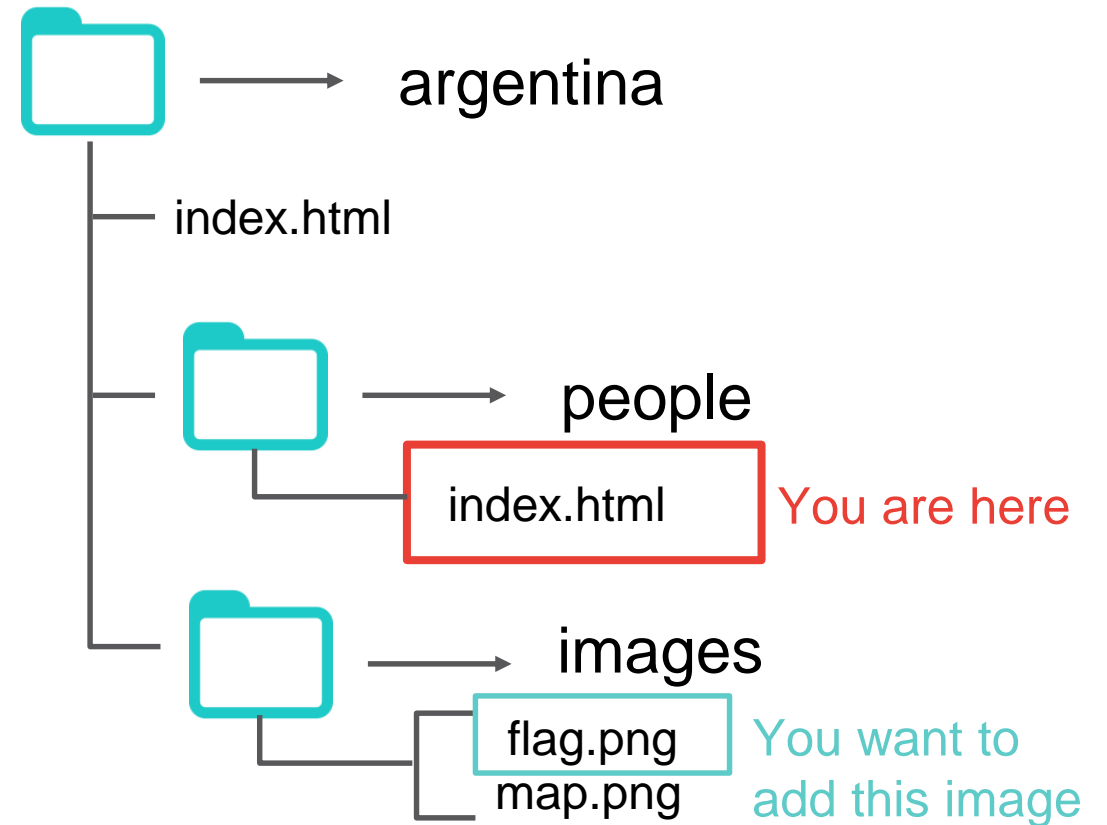
	DESCRIPTION	EXAMPLE
SAME	File name	info.html
CHILD	Name of child folder + / + file name	people/index.html
GRANDCHILD	Name of child folder + / + Name of grandchild folder + / + file name	people/culture/index.html
PARENT	../ + path	../index.html
GRANDPARENT	../.. / + path	../.. /index.html

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

RELATIVE URLS

1. First we use ../ to go back to the parent directory
2. Then we add the path to the image

```
img src="../../../images/flag.png" alt="Flag">
```



ACTIVITY



EXERCISE

KEY OBJECTIVE

- Practice writing absolute and relative paths

TIMING

5 min

1. Work with partner to answer questions 1-4

5 min

2. Selected pairs will share with rest of group

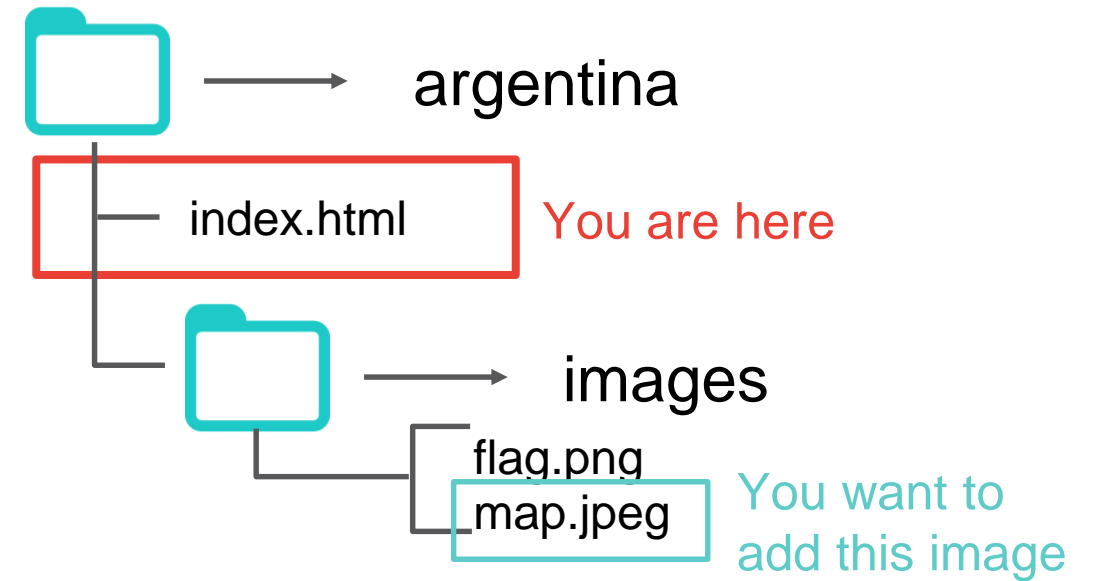
QUESTION 1

- We want to add a link from our homepage (index.html) to our about page (about.html)
- Write out the anchor tag (including href attribute)



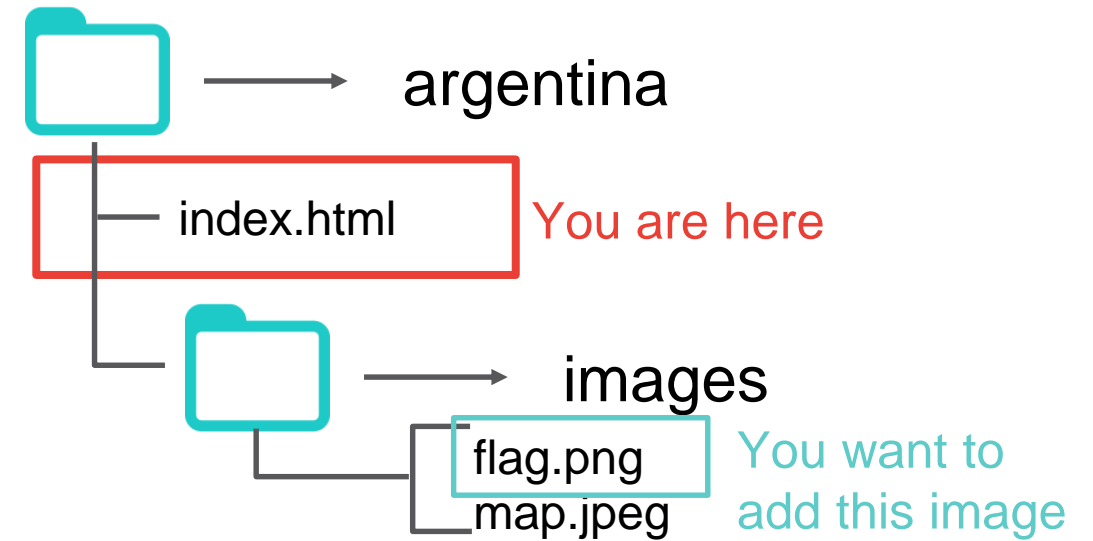
QUESTION 2

- We want to add the image map.jpeg to our homepage (index.html)
- Write out the image tag (include src and alt attributes)



QUESTION 3

- We want to add the image flag.png to our homepage (index.html)
- Write out the image tag (include src and alt attributes)

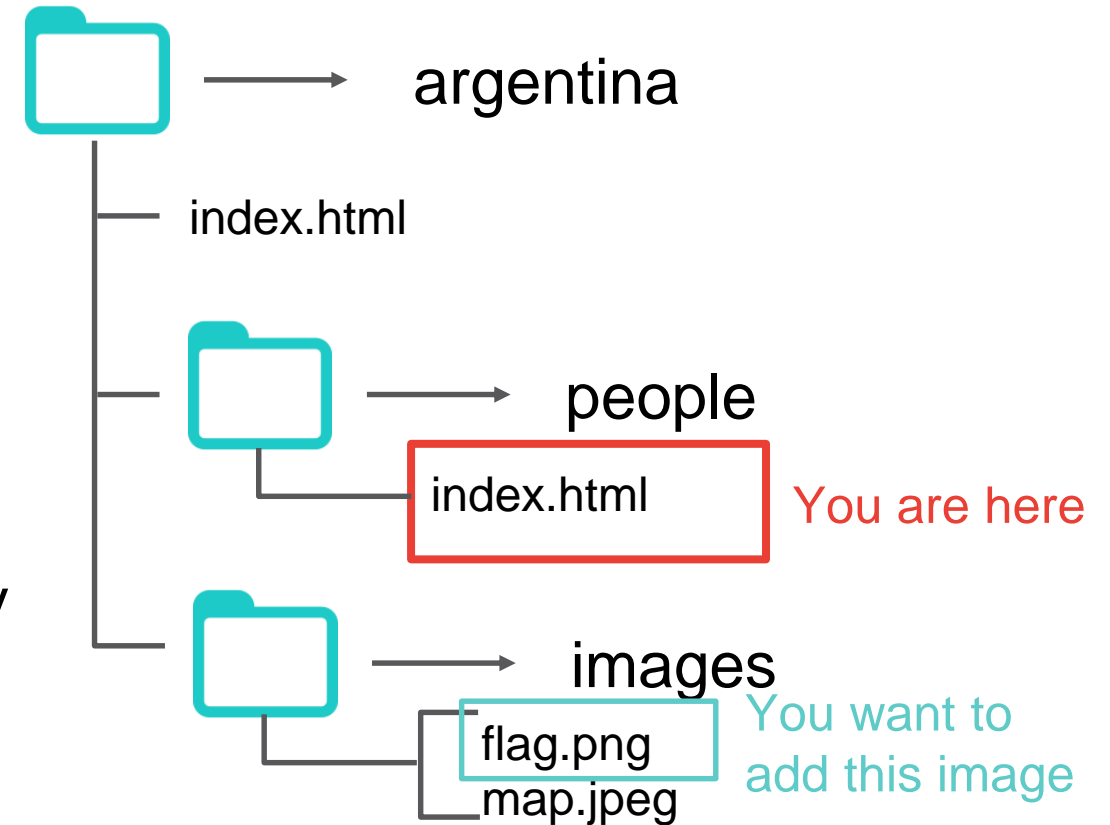


QUESTION 4

- We want to add flag.png to the homepage of the "people" section of our site.
- Write out the image tag (include src and alt attributes)

HINT:

1. First we use ../ to go back to the parent directory
2. Then we add a path to the image



LEARNING OBJECTIVES

- Describe the DOM and draw simple DOM tree.
- Predict image paths and apply relative paths to and <a> tags.
- Differentiate between the various image file types
- Apply and explain CSS “cascade” including: importance, specificity and inheritance.
- Differentiate between basic web color principles: RGB, RGBA, hexadecimal color, HSL.

IMAGE FORMATS: JPEG

- ▶ JPEGs should be used for images with many different colors in the picture.



IMAGE FORMATS: GIF AND PNG

- Use GIF or PNG with saving images with few colors (a.k.a flat color)

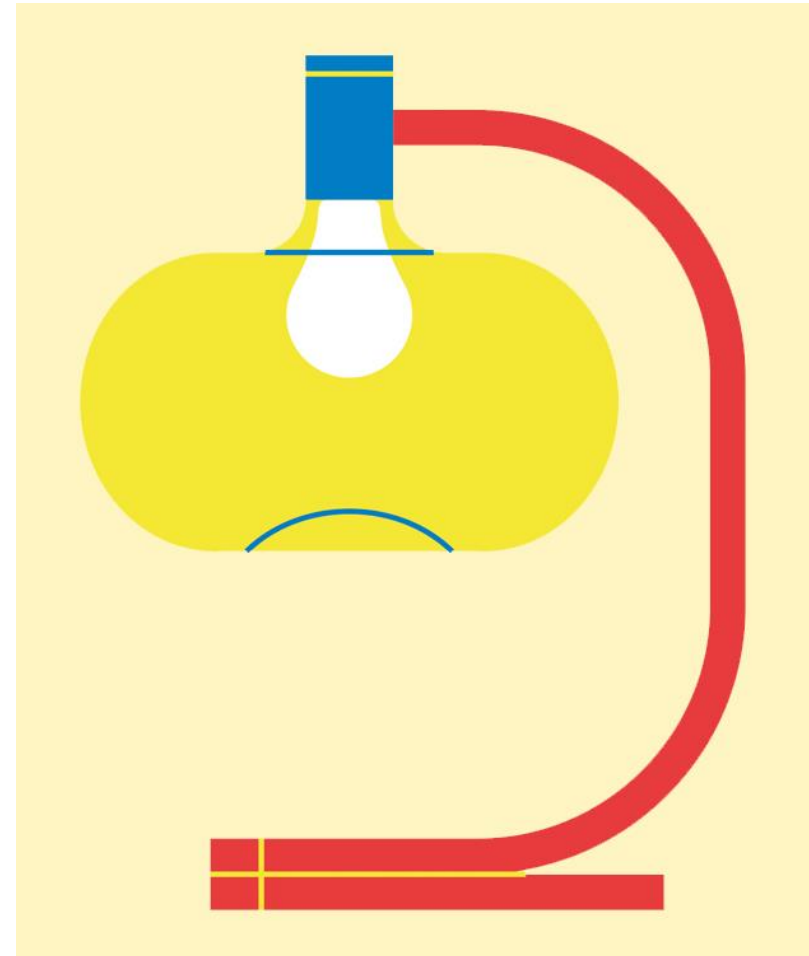


IMAGE FORMATS: PNG

- Supports transparency and semi-transparency, great for logos, icons, and repeating background tiles.
- Supports semi-transparency
- Almost always preferable to a gif, unless semi-transparency is not needed, and the gif format is significantly smaller.

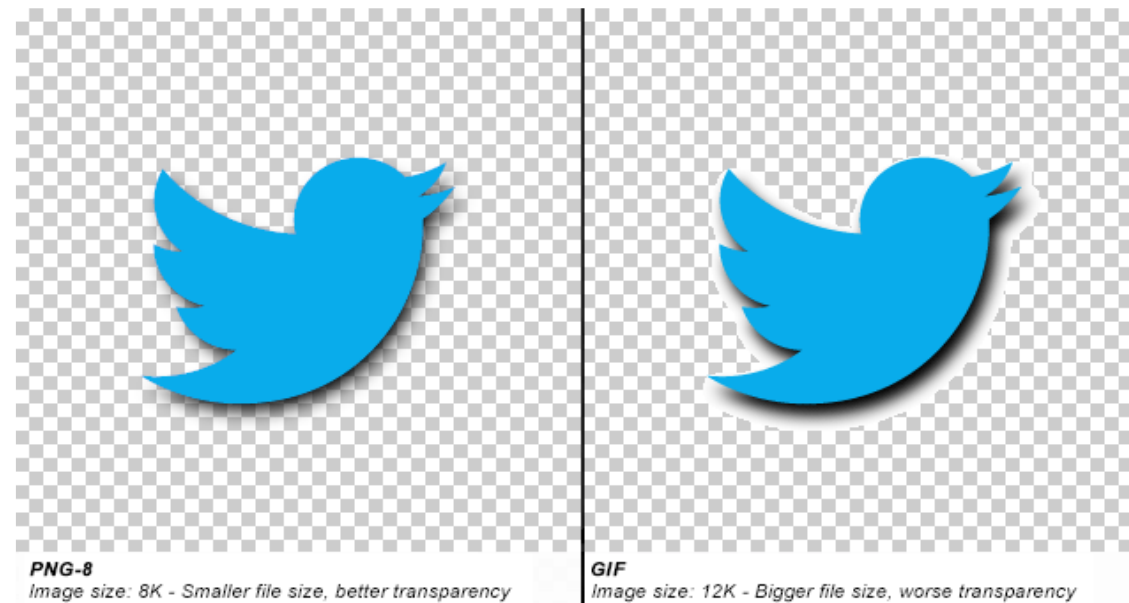


IMAGE FORMATS: GIF

- Can have basic transparency, typically a png is used instead.



ACTIVITY



EXERCISE

KEY OBJECTIVE

- Identify which file format should be used for selected images

TIMING

- | | |
|-------|---|
| 2 min | 1. Work with a partner to answer questions 1-6 |
| 2 min | 2. Selected pairs will share with rest of group |

ACTIVITY

1.



2.



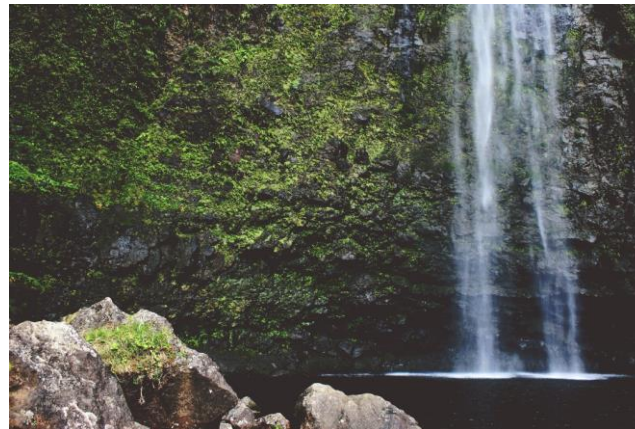
3.



4.



5.



6.



FEWD – LESSON 2

INTRO TO CSS

LEARNING OBJECTIVES

- Describe the DOM and draw simple DOM tree.
- Predict image paths and apply relative paths to and <a> tags.
- Differentiate between the various image file types
- Apply and explain CSS “cascade” including: importance, specificity and inheritance.
- Differentiate between basic web color principles: RGB, RGBA, hexadecimal color, HSL.

WHAT IS CSS?

- The key to understanding how CSS works is to imagine that there is an invisible box around every HTML element
- CSS allows you to create rules that control the way that each individual box (and the contents of that box) is presented

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.

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.

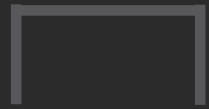
WHAT IS CSS?

WHAT DOES CSS DO?

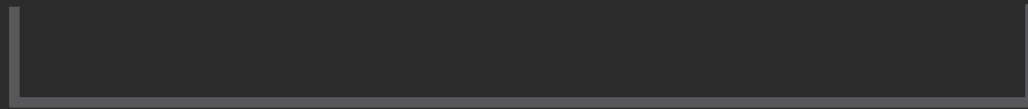
- CSS is what handles the presentation layer of our webpage.
- It allows us to associate style rules with HTML elements.

CSS SYNTAX

Selector



```
h1 { color: yellow; }
```



Declaration

CSS SYNTAX

h1, h2 {

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

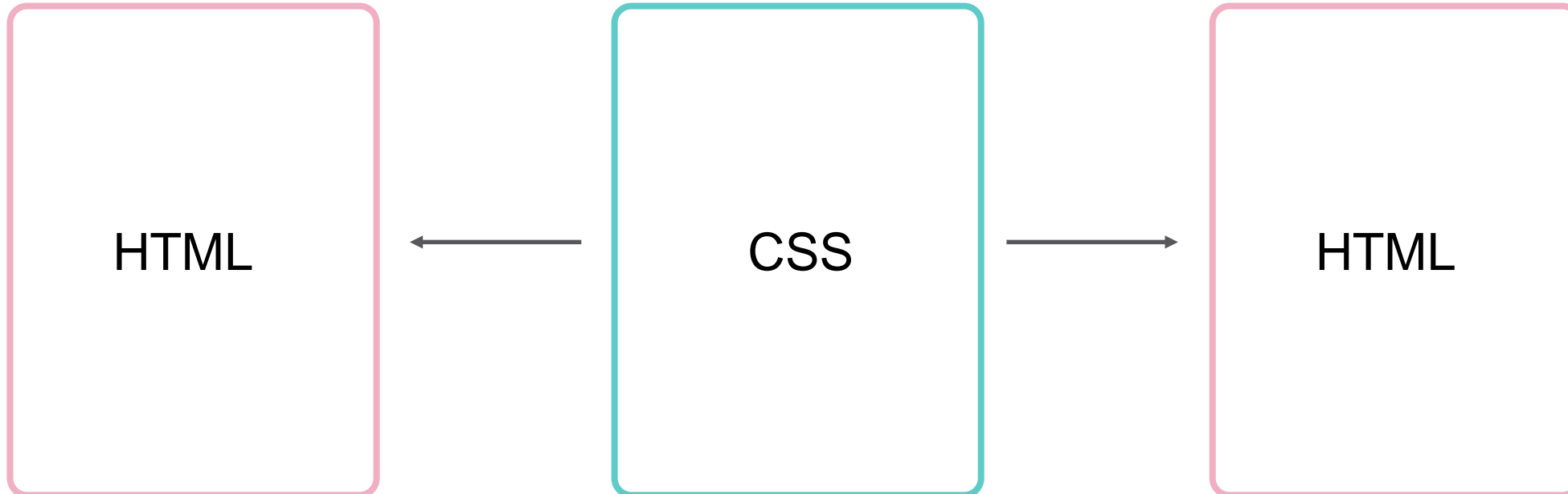
- Use the <link> element to tell the browser where to find the CSS file that should be used to style the page.
- This should live inside the <head> of the document.

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

- The href specifies the path to the CSS file.
- The rel attribute specifies the relationship between the HTML page and the file it is linked to.

INTERNAL VS. EXTERNAL CSS

- You should use an external CSS stylesheet whenever possible
- This allows multiple pages to use the same stylesheet (Don't repeat yourself!)
- Keeps the content separate from the presentation



COOKING RECIPE CODE ALONG – PART 1



Let's code! Drag the folder `cookie_recipe_css` into sublime text

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.
- It's important to understand which rule will take precedence when there are two or more rules that apply to the same element.

LAST RULE

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

SPECIFICITY

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

IMPORTANT

- You can add !important after any property value to indicate that it should be considered more important than other rules that apply to the same element.

INHERITANCE

- Some properties are inherited by child elements.
- If you specify a font-family for the body, it will be inherited by child elements.
- You can compare this with the background-color or border property, these will not be inherited by child elements.

LEARNING OBJECTIVES

- Describe the DOM and draw simple DOM tree.
- Predict image paths and apply relative paths to and <a> tags.
- Differentiate between the various image file types
- Apply and explain CSS “cascade” including: importance, specificity and inheritance.
- Differentiate between basic web color principles: RGB, RGBA, hexadecimal color, HSL.

COLOR

TEXT COLOR:

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.

```
h1 {  
  color: blue;  
}  
  
p {  
  color: red;  
}
```

BACKGROUND COLOR:

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.

```
body {  
  background-color: blue;  
}
```

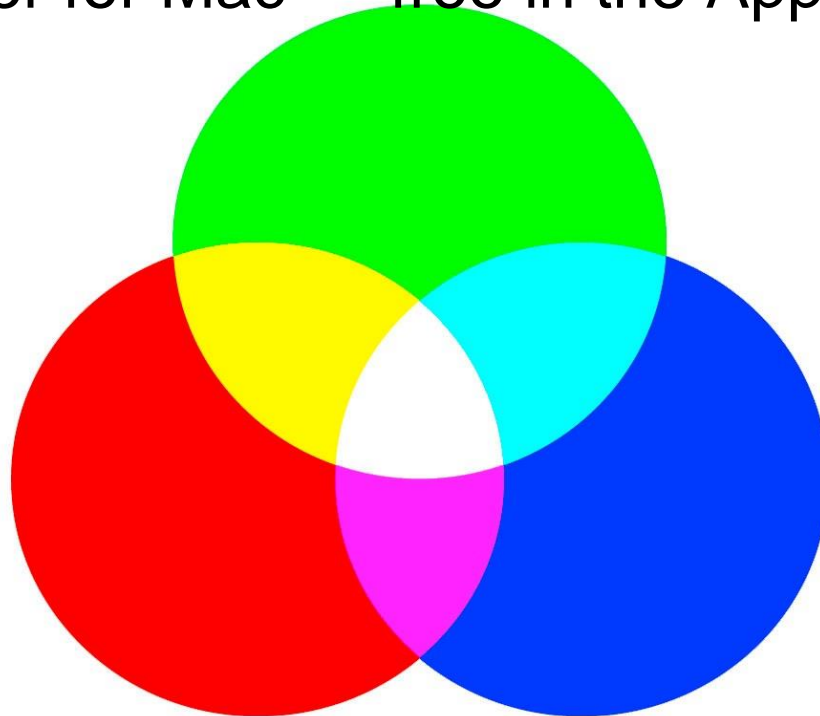
COOKING RECIPE CODE ALONG – PART 2



Let's code! Drag the folder `cookie_recipe_css` into sublime text

COLOR

- Every color on a computer screen is created by mixing amounts of red, green, and blue
- To find the color you want, you can use a color picker
- Color pickers are available in image editing programs like Photoshop.
- You can also use the [Sip](#) color picker for Mac — free in the App Store.



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

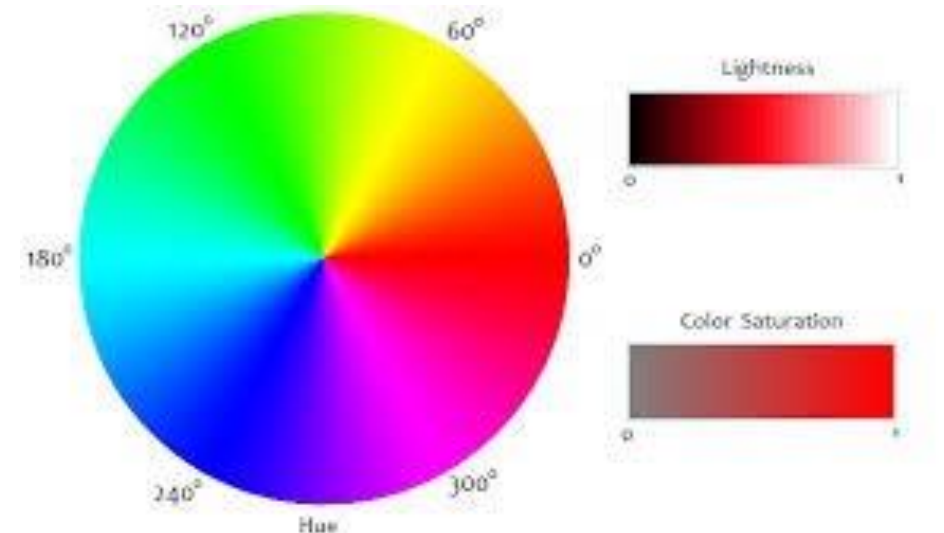
COLOR

HSL

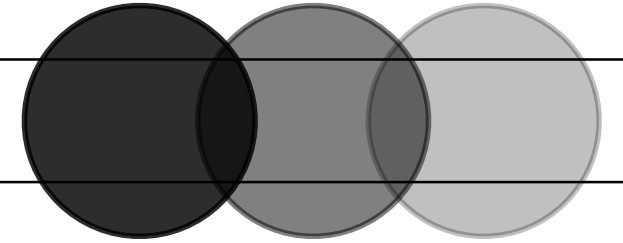
- Similar notation to RGB values, but specify colors using hue, saturation, and lightness.
- Hue: degree angle measure. Red at 0, green at 120, and blue at 240.
- Saturation: percentage. 100% is fully saturated, 0% is a shade of grey
- Lightness: percentage. 0% — black, 100% — white, 50% — normal



hsl(178, 59.8%, 55.1%)

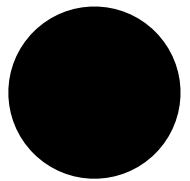


OPACITY

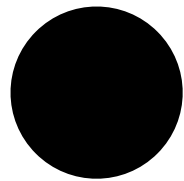


RGBA AND HSLA

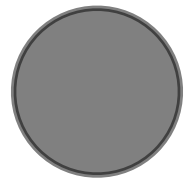
- RGBA and HSLA work the same as RGB and HSL, except that they take a 4th value called 'alpha'.
- This is a value between 0 and 1 which can be used to determine a color's opacity on the page.



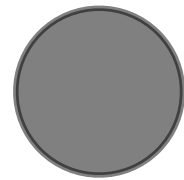
`rgba(0, 0, 0, 1)`



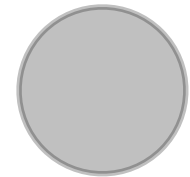
`hsla(0, 0%, 0%, 1)`



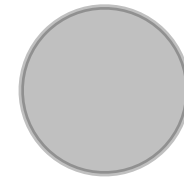
`rgba(0, 0, 0, 0.5)`



`hsla(0, 0%, 0%, 0.5)`



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



`hsla(0, 0%, 0%, 0.25)`

TYPEFACE TERMINOLOGY

TEXT-ALIGN

left
center
right
justify

TEXT-TRANSFORM

UPPERCASE
lowercase
Capitalize

TEXT-DECORATION

none
underline

LINE-HEIGHT

20px

FONT-WEIGHT

Normal
Bold

FONT-STYLE

Regular
Italic

FONT-FAMILY

sans-serif
serif

FONT-SIZE

12px
26px

FONT-FAMILY

serif

- Georgia
- Times
- Times New Roman

sans-serif

- Arial
- Verdana
- Helvetica



FEWD – LESSON 2

LAB TIME!!!

**ABOUT ME
LANDING PAGE**

CREATING A NEW REPOSITORY – THANKS EMILY!

1. Navigate to <https://github.com/> and login.
2. Select “New Repository”
3. Name the Repository (no spaces) and click “Create repository”
4. Select “Set up in Desktop”
5. Specify where you want the repository folder
6. Navigate to your files and drag them to the newly created repository folder.
7. Navigate to GitHub Desktop
8. On the “Changes” tab, add a summary of what was updated/added to the folder.
9. Select “Commit to master”
10. Publish
11. Navigate to <https://github.com/> and select your repository to see files listed.
12. Use provided link (in URL or on the right) to share with others.

HTML BASICS

HTML5 STARTER TEMPLATE

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
  <head>
```

```
    <meta charset="UTF-8">
```

```
    <title>Document</title>
```

```
  </head>
```

```
  <body>
```

```
  </body>
```

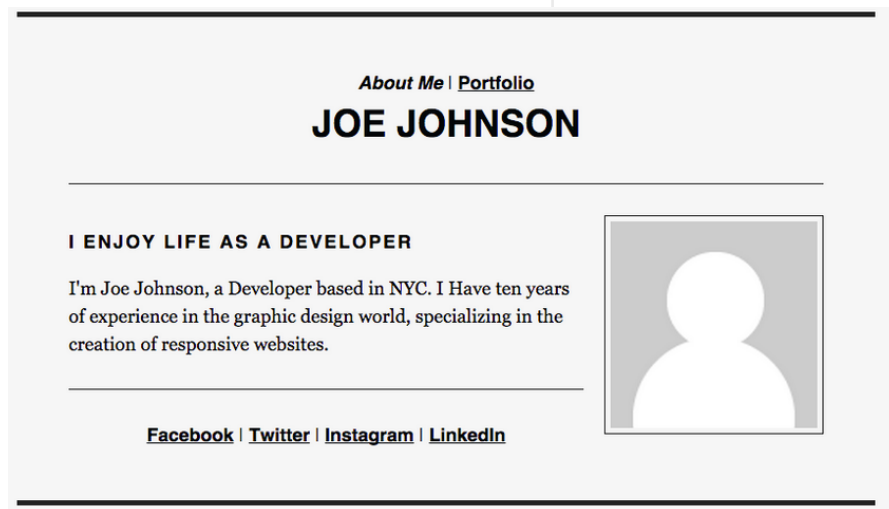
```
</html>
```

 **Your code here**

ASSIGNMENT 1 – ABOUT ME LANDING PAGE

KEY OBJECTIVE

- Clearly articulate your skills and experience on a landing page
- Style a page using CSS via an external stylesheet
- Practice using GitHub to manage a project



REQUIREMENTS

- Use a unique title and a single unique h1 tag
- Show images using img tags including unique alt attributes
- Use an external CSS stylesheet to style your pages
- Select at least 5 elements on the page and apply at least 5 different css properties (font-size, color, font-family, background, text-decoration) to these elements
- Use at least 6 different HTML tags (h1, h2, h3, h4, h5, h6, p, ul, ol, a, img) on your pages
- Indent nested elements to increase your code's readability

FEWD – LESSON 2

WRAP UP

LEARNING OBJECTIVES

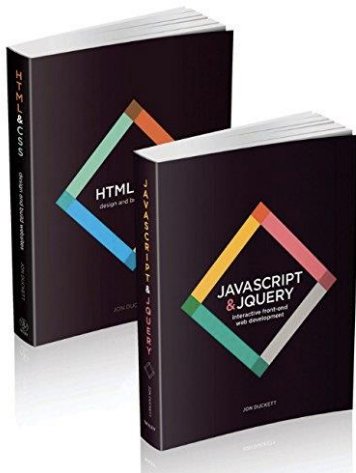
- Describe the DOM and draw simple DOM tree.
- Predict image paths and apply relative paths to and <a> tags.
- Differentiate between the various image file types
- Apply and explain CSS “cascade” including: importance, specificity and inheritance.
- Differentiate between basic web color principles: RGB, RGBA, hexadecimal color, HSL.

HOMEWORK

- Finish your about me landing page! Please email a link to your repository to daveturner81@gmail.com before 11:59pm on Sunday. If you're struggling with GitHub you can also send a zip file

RECOMMENDED COURSE MATERIALS:

- [Jon Duckett - Web Design with HTML, CSS, JavaScript and jQuery Set](#)



FEWD – LESSON 2

EXIT TICKETS

[Click Here](#)

Course/Cohort: **CHI - FEWD5**

Lesson number: **2**

Topic: **CSS Basics**