

FEWD Week 1 • Class 2:
CSS Building Blocks





REVIEW: BUILDING BOUNTY20

OBJECTIVES

- Linking to external files
- Using different fonts on the web
- Revisiting the box model
- Units of measurement on the web
- Working with images
- Colors on the web

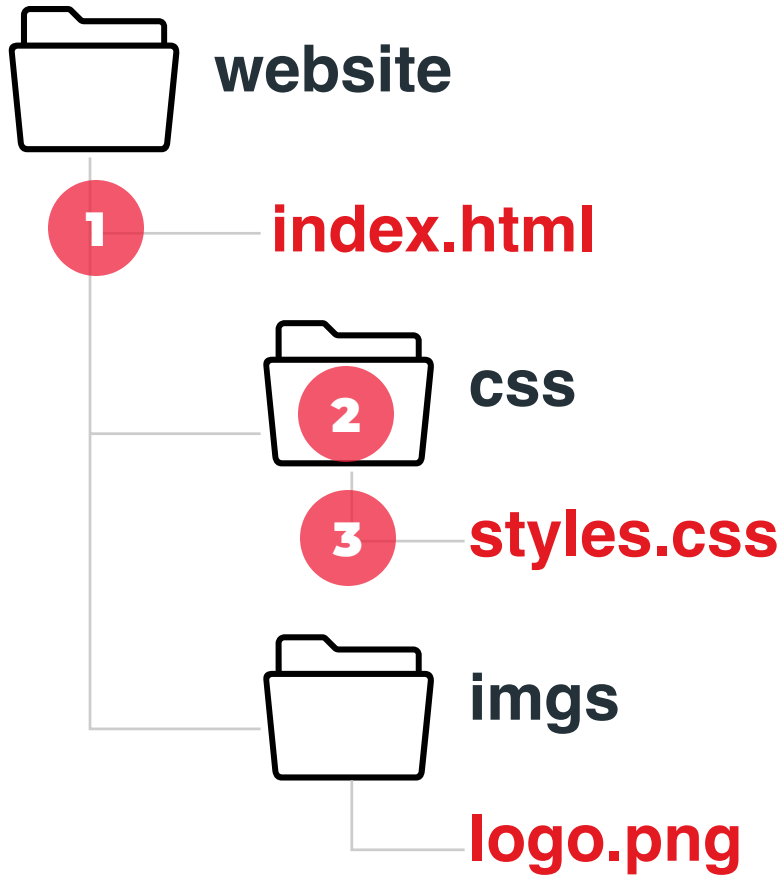
EXTERNAL STYLESHEETS

LINKING FILES

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

🚩 **HEADS UP:** No `<style>` tags are used!

LINKING FILES RELATIVELY



index.html → styles.css:

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

- 1** Start in the folder where the current file is
- 2** Go into the folder called **css**
- 3** Get the file called **styles.css**



EXTERNAL CSS

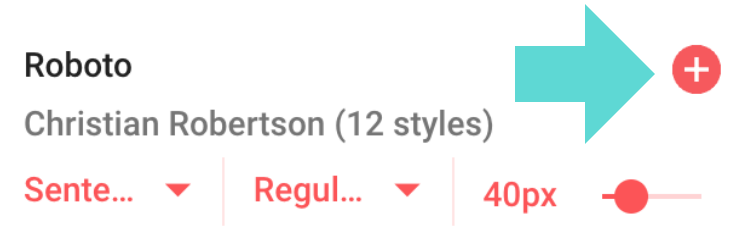
GOOGLE FONTS

USING DIFFERENT FONTS

- We can't be certain which fonts our visitors have, so we need to send the fonts with our site files.
- Font files are **big** and can slow down the load time of your site.
- The Google Fonts service delivers fonts via a **super fast collection of servers** called a CDN.
- CDNs check whether a file has been previously downloaded before sending it again.

LOADING GOOGLE FONTS

1. Go to <https://fonts.google.com>.
2. Choose a font and click the ⊕.
3. In the popup at the bottom of the screen, click the **customize** tab to select the font weights you want to use.
4. Next, click back on the **embed** tab and copy the **link** tag.
5. Paste the link tag in your HTML page head tag **before** your linked css file.
6. Back on Google Fonts, copy the css for the font-family to use in your css file!





GOOGLE FONTS

UNITS OF MEASUREMENT

UNITS OF MEASURE

- **px**: A fixed (aka absolute) value in pixels
- **em**: Relative to the font-size of the element
(e.g., 2em = 2 x the size of the current font)
- **rem**: Relative to the root element font-size
- **vh**: % of the viewport height
(e.g., 50vh = 50% of the viewport height)
- **vw**: % of the viewport width
- **vmax**: % of viewport's larger dimension
- **vm**: % of viewport's smaller dimension
- **%**: It depends



UNITS OF MEASUREMENT

CSS BOX MODEL

CSS BOX MODEL

1

The **content box** is defined by the height and width of the elements inside it.

2

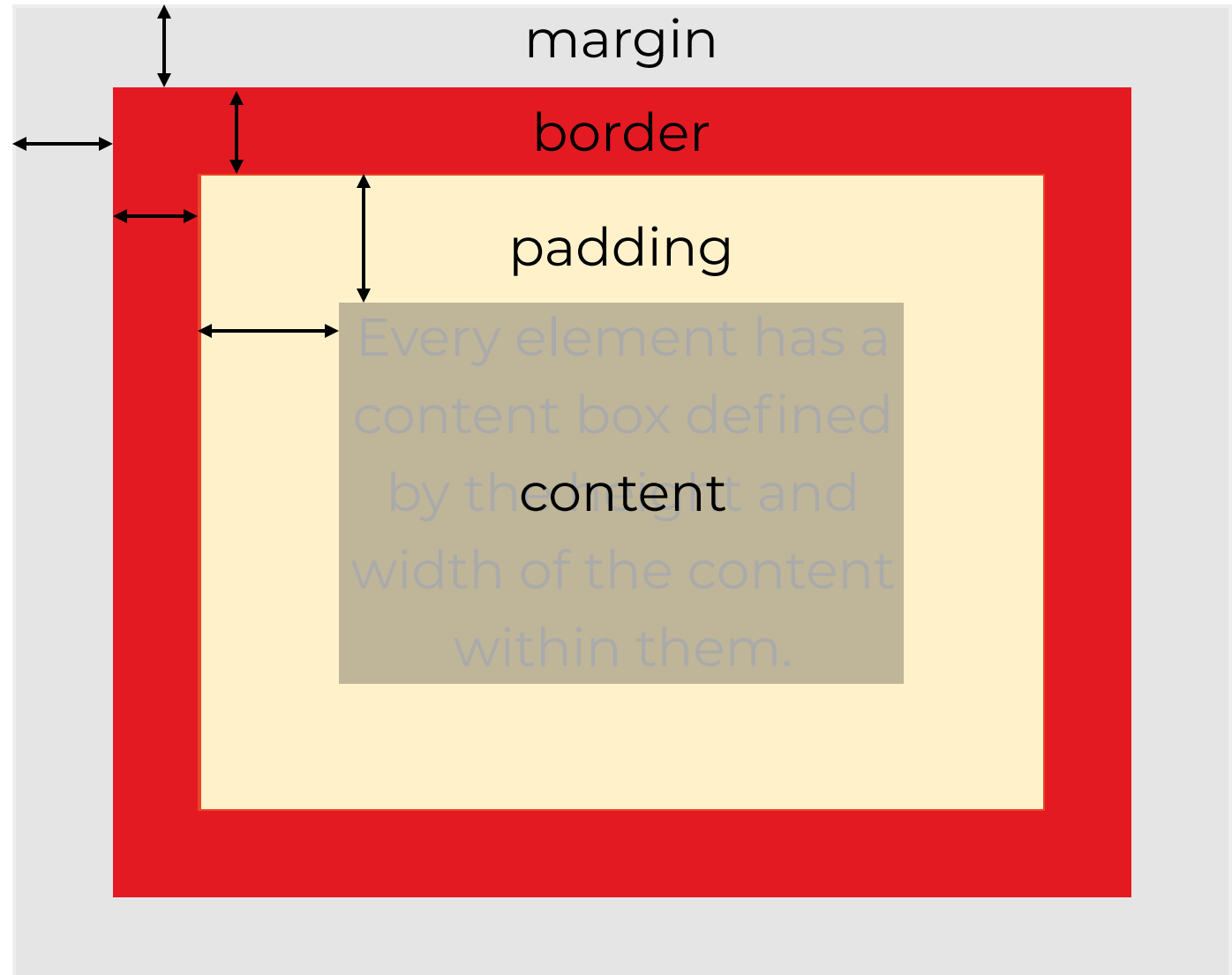
Adding **padding** creates interior space around the content.

3

The background extends to the **border**.

4


The **margin** creates space on the outside of the element.



MARGINS & PADDING REFRESHER

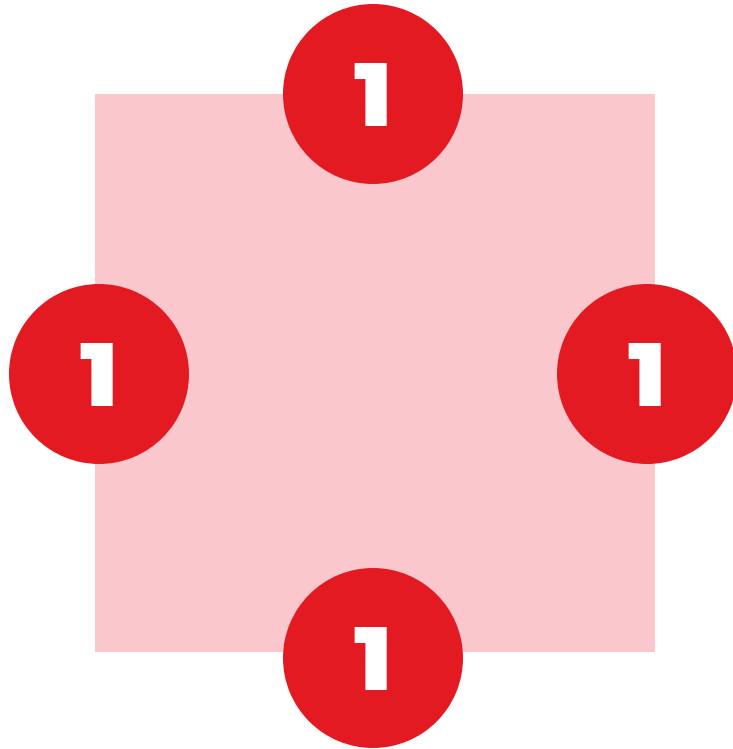
```
div {  
  padding-top: 16px;  
  padding-right: 30px;  
  padding-bottom: 16px;  
  padding-left: 30px;  
}
```

```
div {  
  margin-top: 24px;  
  margin-right: 16px;  
  margin-bottom: 24px;  
  margin-left: 16px;  
}
```

 **HEADS UP:** Margins can have **negative** values.
Padding can only have positive values.

SHORTHAND VERSION

with 1 value

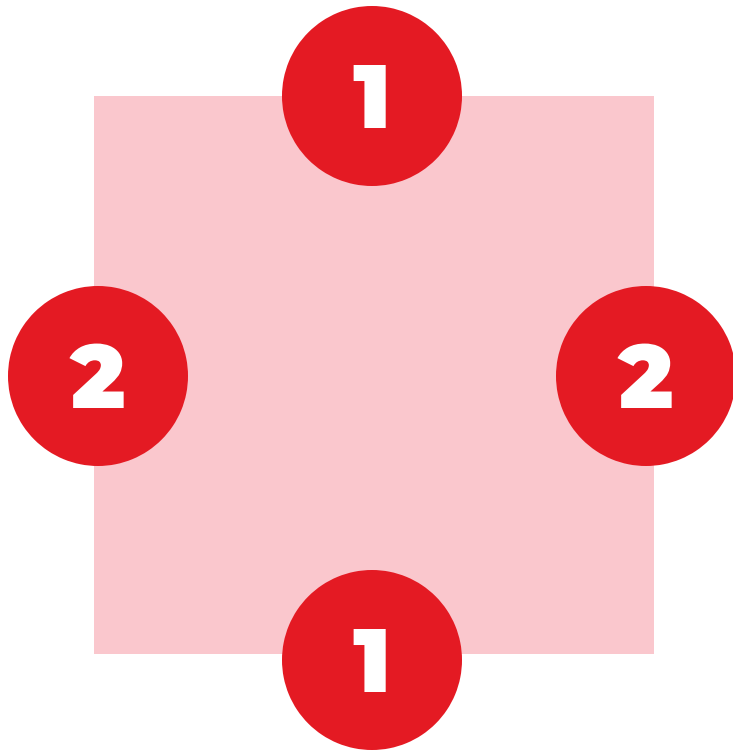


```
div {  
  margin: 20px;  
  padding: 20px;  
}
```

All sides are the same.

SHORTHAND VERSION

with 2 values



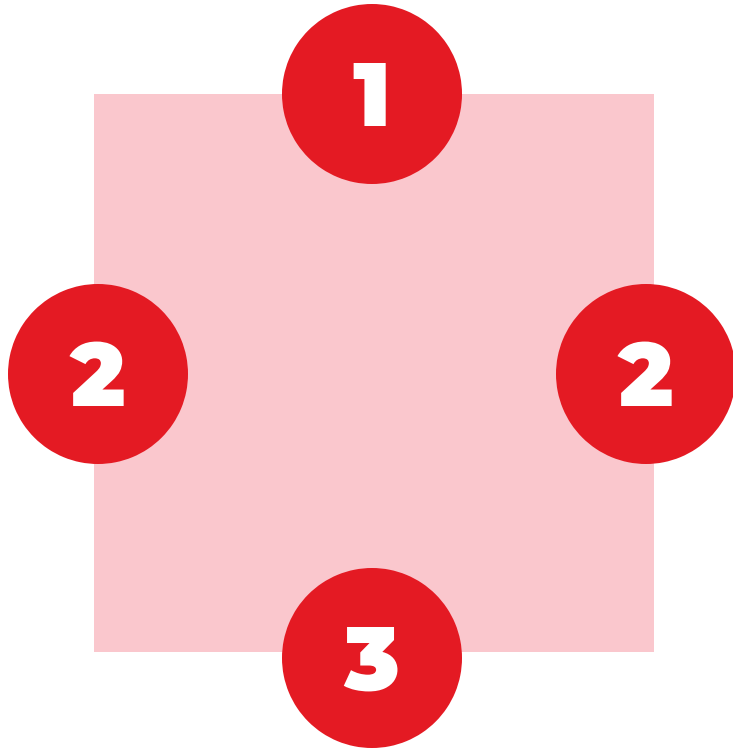
```
div {  
  margin: 1em 20px;  
  padding: 1em 20px;  
}
```

Top/Bottom are 1em

Right/Left are 20px

SHORTHAND VERSION

with 3 values



```
div {  
  margin: 0 1rem;  
  padding: 20px 50px 10px;  
}
```

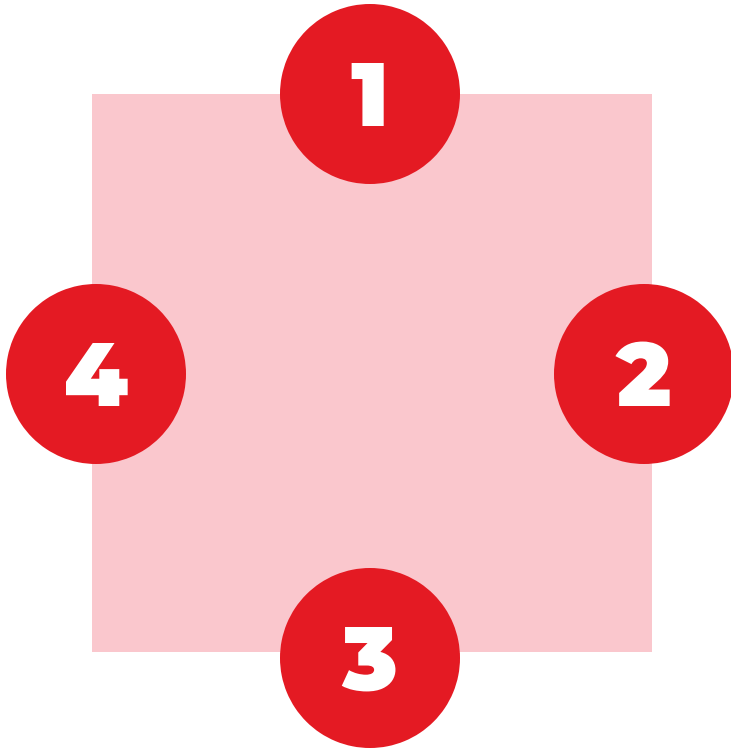
Top padding is 20px

Right/Left padding is 50px

Bottom padding is 10px

SHORTHAND VERSION

with 4 values



```
div {  
  margin: 0 0 1rem 1rem;  
}
```

Trouble remembering?

Top - **R**ight - **B**ottom - **L**eft

HEIGHT & WIDTH

Height and width can only be applied to **block** elements.

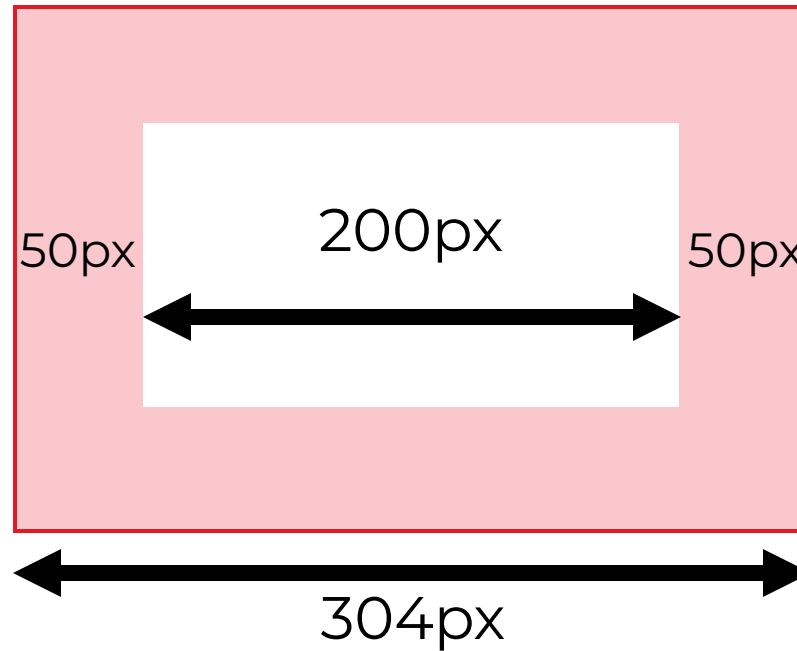
```
div {  
    height: 50vh;  
    width: 100%;  
}  
span {  
    height: 50px; /* HAS NO EFFECT */  
}
```

APPLYING HEIGHT & WIDTH

- Height and width are applied to the size of the **content box** by default.
- If the element has borders, padding and margin, that is added to the height and width values provided.
- We can fix this by setting the `box-sizing` property in css to border-box.

BOX-SIZING: CONTENT-BOX

```
div {  
  border: 2px solid;  
  padding: 50px;  
  width: 200px;  
  box-sizing: content-box;  
}
```



CONTENT-BOX (DEFAULT)

PADDING LEFT: 50PX

PADDING RIGHT: 50PX

BORDER LEFT: 2PX

BORDER RIGHT: 2PX

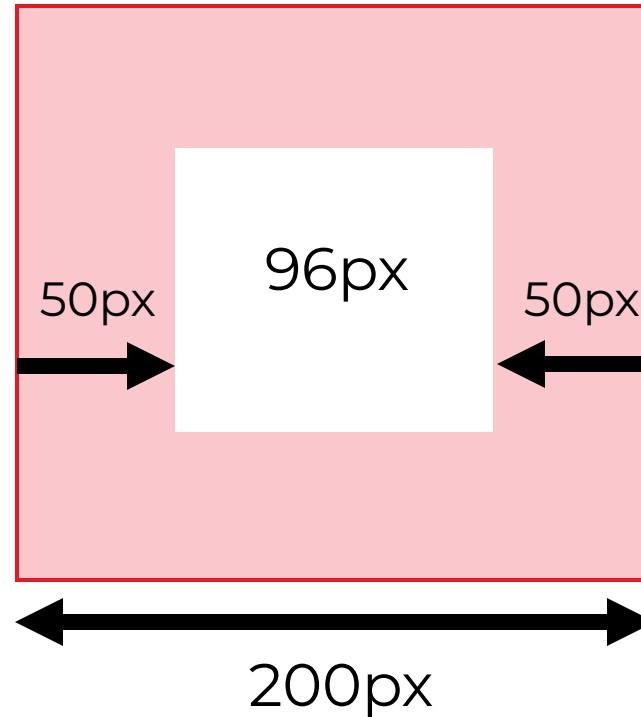
+ WIDTH (CONTENT): 200PX

TOTAL SIZE: 304PX

🚩 **HEADS UP:** content-box is the default value.

BOX-SIZING: BORDER-BOX

```
div {  
  padding: 50px;  
  border: 2px solid;  
  width: 200px;  
  box-sizing: border-box;  
}
```



BORDER-BOX

WIDTH (TOTAL): 200PX

- BORDER-LEFT: 2PX

- BORDER-RIGHT: 2PX

- PADDING LEFT: 50PX

- PADDING RIGHT: 50PX

CONTENT: 96PX

🚩 **HEADS UP:** border-box is the preferred value.

FIXING THE BOX MODEL

We use a **reset** to apply the box-sizing to everything with the `*` universal selector.

```
/* This special selector means apply this to everything! */  
  
* {  
  box-sizing: border-box;  
}
```

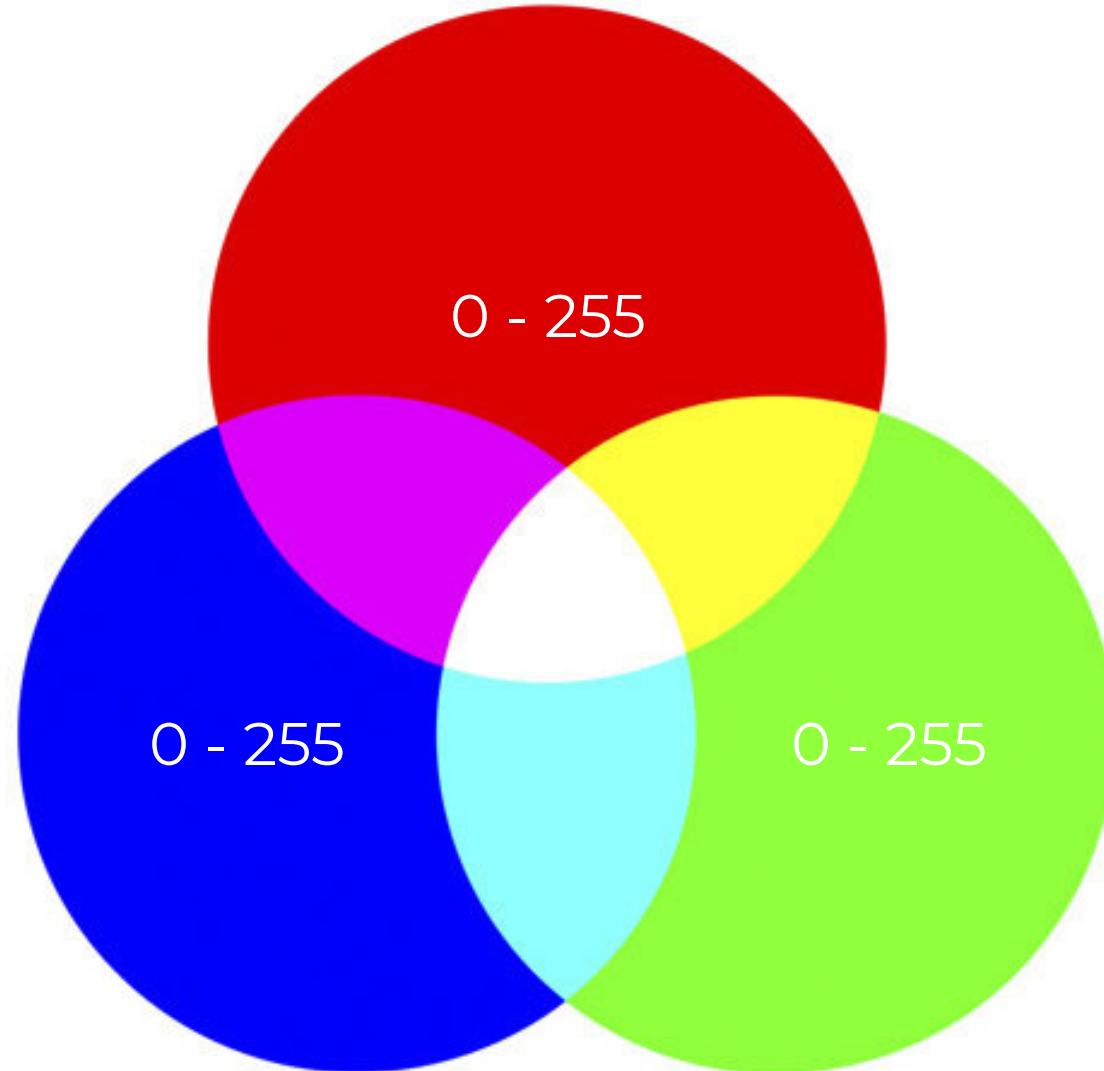


BOX-SIZING, HEIGHTS & WIDTHS, AND PADDING

COLORS

RED × GREEN × BLUE

16,777,216 possible color combinations



COLOR VALUES IN CSS

- Keyword
- Hexidecimal
- RGB & RGBA
- HSL & HSLA

 **HEADS UP:** Alpha channel range is 0 (transparent) to 1 (opaque)

```
/* Keyword Syntax */

h1 {
  background-color: gray;
}

/* RGB & HSL Syntax */

p {
  color: rgba(0,0,0,1);
  border: 2px solid hsl(0,0%,0%);
}

/* Hexadecimal Syntax */

div {
  background: #ff0000;
}
```



**ADDING COLORS &
TRANSPARENCY**

IMAGES

ADDING IMAGES TO YOUR PAGE

- Images can be added with the `` tag in HTML or in the background via CSS
- Background images are design elements only and are ignored by screen-readers

IMAGE TYPES

- JPG: A **lossy** format that is great for small file sizes of raster images (like photos), but it doesn't have transparency.
- PNG: A **lossless** raster format that has transparency.
- SVG: A vector file format that is great for images that can be drawn.
- GIF: A **lossy** format that can make one color transparent and can store multiple images for animation.

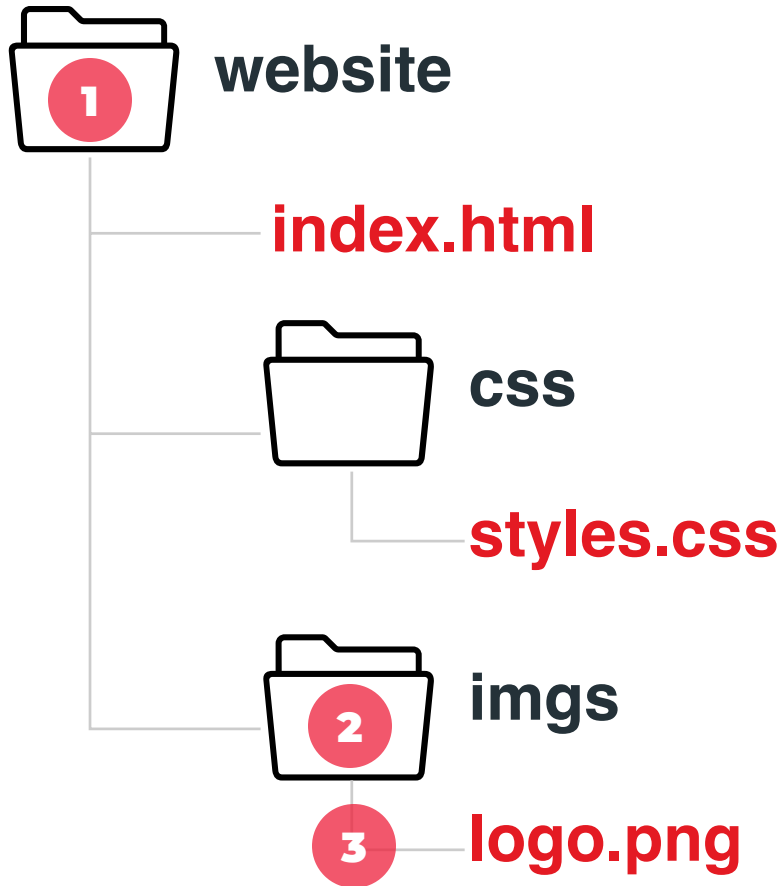
CONTENT IMAGES

- The `` tag has no closing tag
- The `src` attribute links the file and is required
- The `alt` attribute is used by screenreaders and for SEO

```

```

LINKING FILES RELATIVELY



index.html → logo.png:

2 **3**
``

- 1** Start in the folder where the current file is
- 2** Go into the folder called **imgs**
- 3** Get the file called **logo.png**



ADDING AN IMAGE

BACKGROUND IMAGES

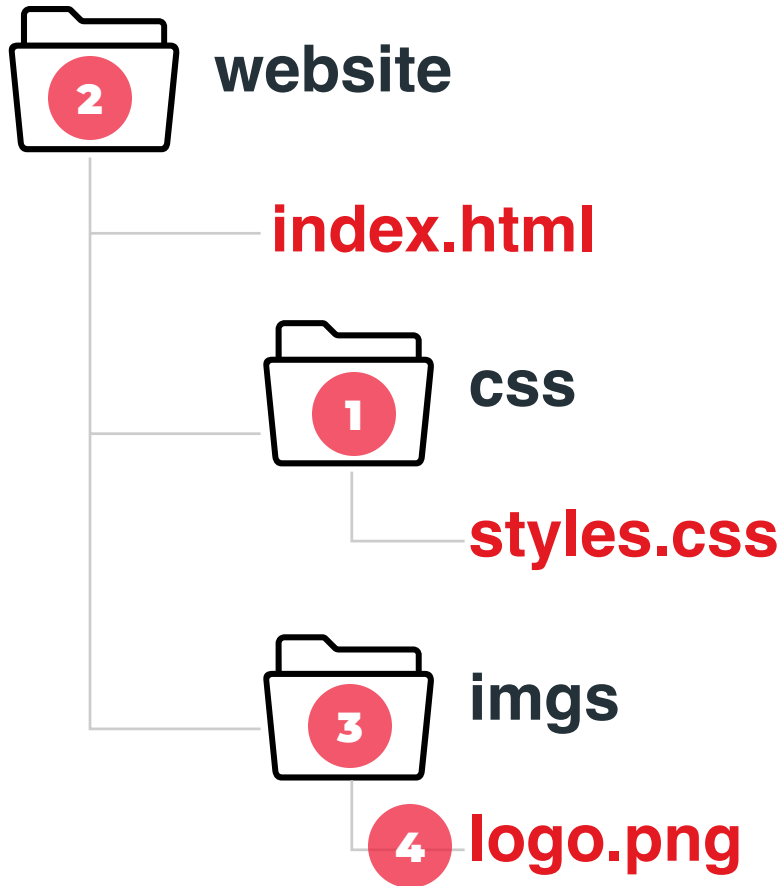
Background images are added through CSS

```
/* Use the background-image or  
   shorthand background property */  
  
selector {  
    background-image: url('path/to/file');  
}
```



HEADS UP: FYI... multiple background images are stackable!

LINKING FILES RELATIVELY



styles.css → logo.png:

```
div {  
    background-image: url("../imgs/logo.png");  
}
```

- 1 Start in the folder where the current file is
- 2 Go back one folder (to **website**)
- 3 Go into the **imgs** folder
- 4 Get the **logo.png** file

BACKGROUND PROPERTIES

- **background-repeat**: tile the image or place once
- **background-position**: position the image in its containing element
- **background-attachment**: scroll with the page or remain fixed in one place
- **background-size**: the size of the image in the background
- **background-clip**: crop the image at the content-box, border-box or padding-box
- **background-origin**: place the image origin at the content-box, border-box or padding-box

BACKGROUND IMAGE SYNTAX

```
header {  
    background-image: url('../images/bg.jpg');  
    background-size: cover;  
}  
  
section { /* Multiple Backgrounds */  
    background-image: url('../images/logo.png'),  
                      url('../images/bg.jpg');  
    background-repeat: no-repeat, repeat;  
    background-position: bottom right, top left;  
}  
  
main { /* Shorthand Format */  
    background: url('../images/logo.png')  
                no-repeat  
                bottom right / 30%  
                fixed;  
}
```



BACKGROUND IMAGES

CLASSES & ID

CSS SELECTORS

1. Element Tags
2. **Classes & IDs**
3. Combinators
4. Attributes
5. Pseudo Classes

CLASSES & IDS

- Classes and IDs give us more flexibility to target elements on our page and apply styles to them
- They also make it easier for us to develop and manage our webpages
- You can combine them with other selectors

IDS

- An ID name is unique. It may only be used once on a page.
- An element may only have one ID

```
<div id="extra-special">
```

```
#extra-special {  
  ...  
}
```

CLASSES

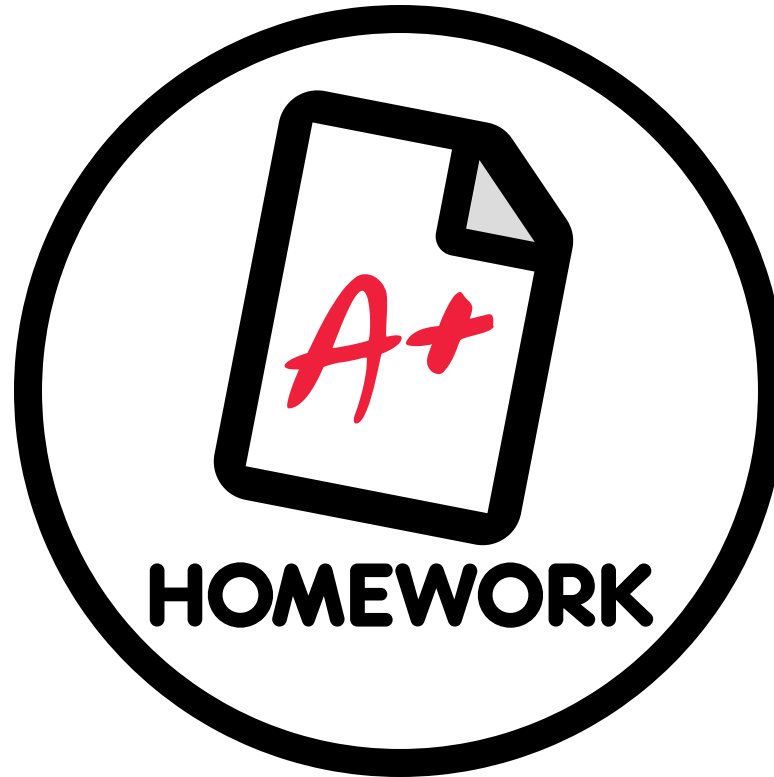
- Classes are reusable as many times as you want
- An element can have as many classes as you want

```
<div class="big primary">
```

```
.primary {  
    ...  
}  
.big {  
    ...  
}
```



CLASSES & IDS



WEEK 1 HOMEWORK

<https://github.com/jmeade11/FEWD/Class2/homework>

HOMEWORK FOR NEXT CLASS

- Complete the tasks to build your personal website.
- Use the Bounty20 website as a guide. All of the techniques are there!
- Use the hints only if you need them. (Try first!)
- Make it your own... I can't wait to see it.

EXIT SURVEY

<https://goo.gl/EB4XFw>

**GO BUILD
AWESOME THINGS!**