



LESSON 04 - CSS LAYOUT

AGENDA

- Learning Objectives
- Review CSS Box Model/CSS Reset
- Divs + HTML5 Structural Elements
- Classes and IDs
- Floats
- Relaxr Lab Time

CLASSROOM CULTURE & VALUES

- Be present and involved
- Be respectful of other people's time
- Honor your commitments
- Be patient!
- Step out of your comfort zone
- Share your Knowledge

LEARNING OBJECTIVES:

AFTER TODAY, YOU SHOULD BE ABLE TO...

- Describe and identify use-cases for normalize.css and reset.css files.
- Draw the DOM Tree for web pages containing `<div>` & `<section>`
- Apply header, footer, sidebar, and multi-column layouts to develop a web page.
- Experiment and predict effects of floats and clearing CSS positioning.

WEEKLY OPEN OFFICE HOURS

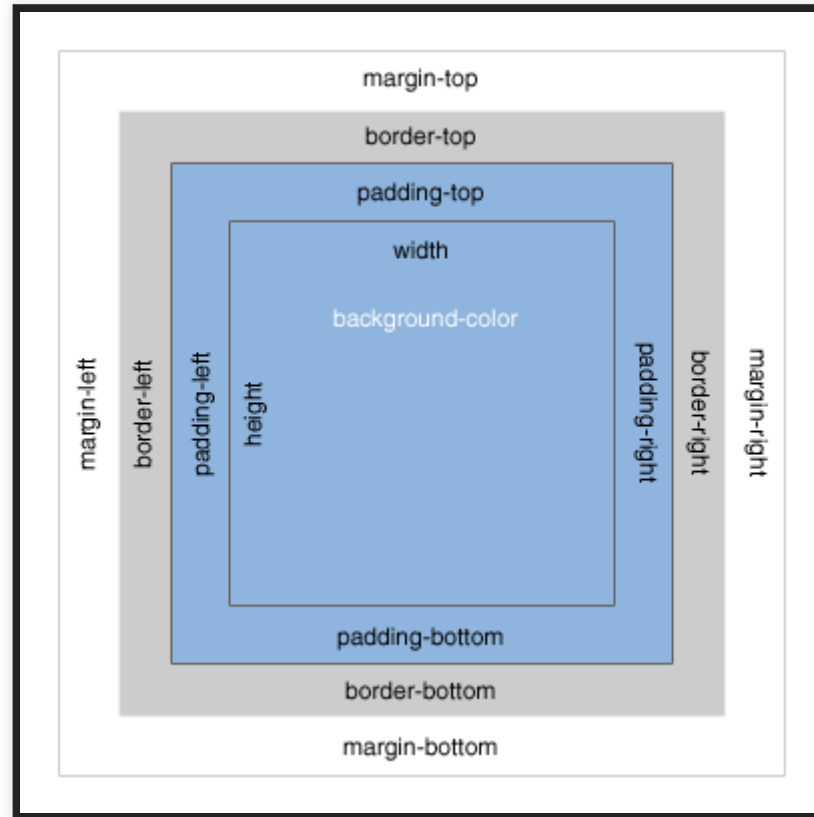
Saturdays between 2pm and 4pm

REVIEW - CSS BOX MODEL

Monday, we learned that each element in an HTML document is a box...

... and this box can be styled using CSS!

These boxes can either fill the full width of the available space, or just the specified space of the content.



Does anyone recall the difference between the box-sizing CSS property values:

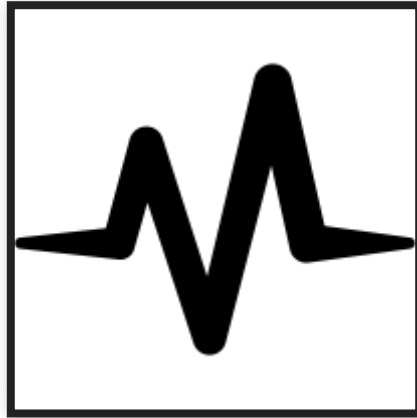
`content-box` and `border-box`

CSS RESET/NORMALIZE

Every browser has its own default 'user agent' stylesheet, that it uses to make unstyled websites appear more legible.

Reset.css/Normalize.css simply clears these styles for
you

This helps you start from a common baseline



PULSE CHECK

Let's compare a sample page with no styles and one with `reset.css`...

The Best Chocolate Chip Cookies



Recipe by: My Grandma

Prep Time

45 Min

Ingredients

- 1 1/2 cup (3 sticks) butter, softened
- 1 cup brown sugar
- 1 cup granulated sugar
- 1 Tbl vanilla instant pudding powder
- 2 Tbl milk
- 2 Tbl vanilla extract
- 2 eggs
- 4 cups all purpose flour
- 2 tsp baking soda
- 1/2 tsp salt
- 4 cups chocolate chips
- 1 cup chopped walnuts or pecans (optional)

How to make Mama's Recipe: The Best Chocolate Chip Cookies

1. Preheat oven to 350 degrees

Now compare that to this version of the same page
with the reset.css file linked to it...

The Best Chocolate Chip Cookies



Recipe by: My Grandma

Prep Time

45 Min

Ingredients

1 1/2 cup (3 sticks) butter, softened

1 cup brown sugar

1 cup granulated sugar

1 Tbl vanilla instant pudding powder

2 Tbl milk

2 Tbl vanilla extract

2 eggs

4 cups all purpose flour

2 tsp baking soda

1/2 tsp salt

4 cups chocolate chips

1 cup chopped walnuts or pecans (optional)

How to make Mama's Recipe: The Best Chocolate Chip Cookies

Preheat oven to 350 degrees

Beat butter and sugars together until light and fluffy

Stir in pudding mix milk and vanilla extract.

Beat in eggs.

Add dry ingredients and stir until combined.

Stir in chocolate chips and nuts.

Place 1 1/2 inch balls of dough 2 inches apart on an un greased cookie sheet.

Bake 8-10 minutes or until golden brown.

Nutrition Information

Probably bad for you, but who cares. MMMMMM COOKIES!!!! nom nom nom

This recipe was altered [from Open Source Recipe](#).

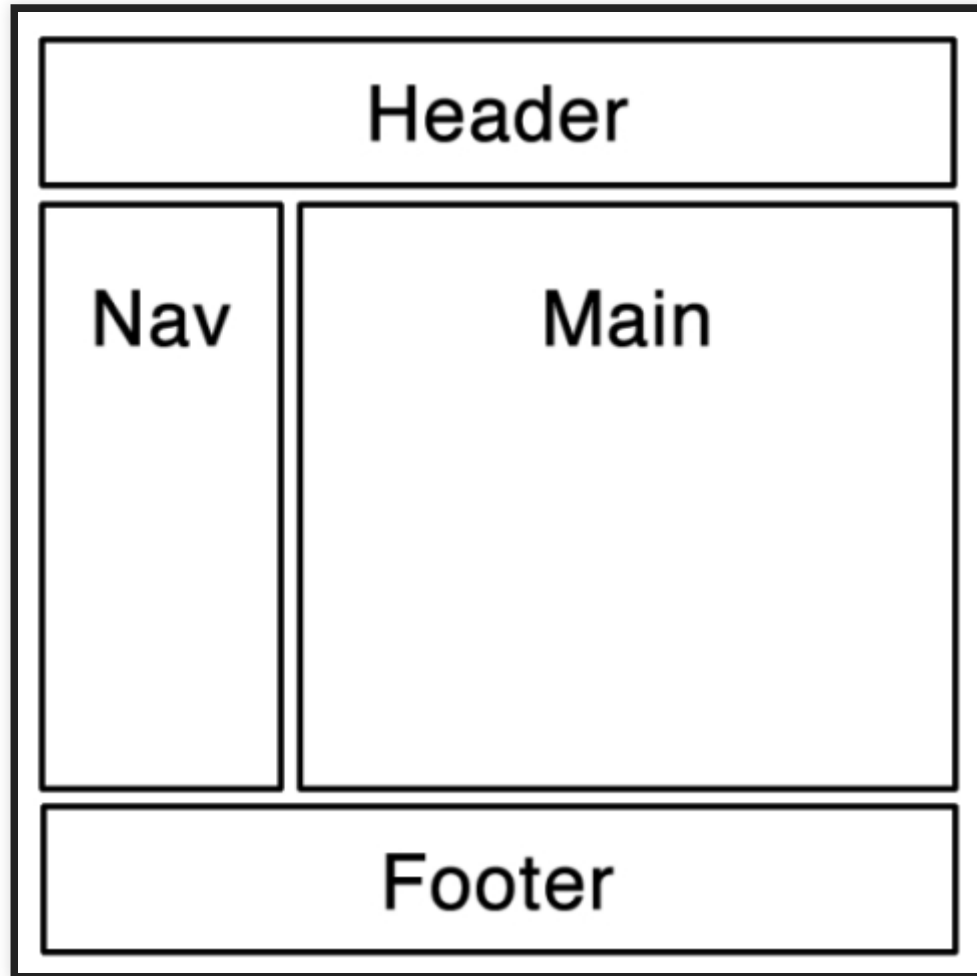
DIV TAG

Let's revisit the `<div>` for a minute...

As mentioned last class, the `<div>` is a block-level element that defines a division or a section in an HTML document.

The `<div>` element has no special meaning at all. It is a generic container element.

Here's a diagram of a traditional 2-column webpage layout:



The HTML markup might look something like this:

```
<div id="header">
    ...
</div>

<div id="nav">
    ...
</div>

<div id="main">
    ...
</div>

<div id="footer">
    ...
</div>
```

As you can see, each section of content is wrapped in its own `<div>`.

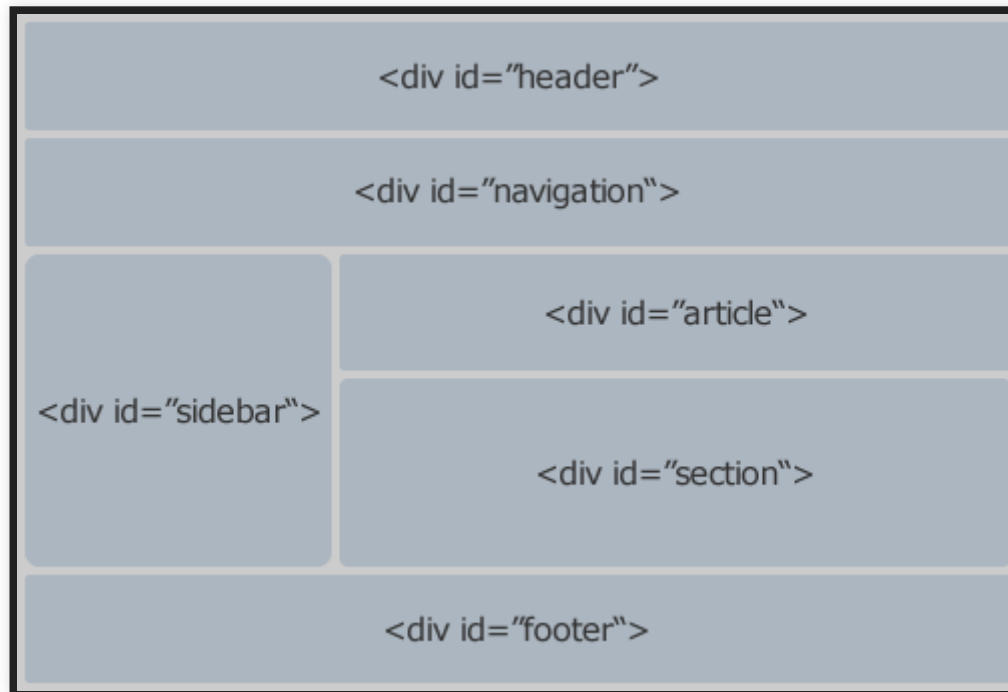
This allows us to style these sections separately.

HTML5 STRUCTURAL ELEMENTS

Prior to HTML5, `<div>` was the only real structural element available.

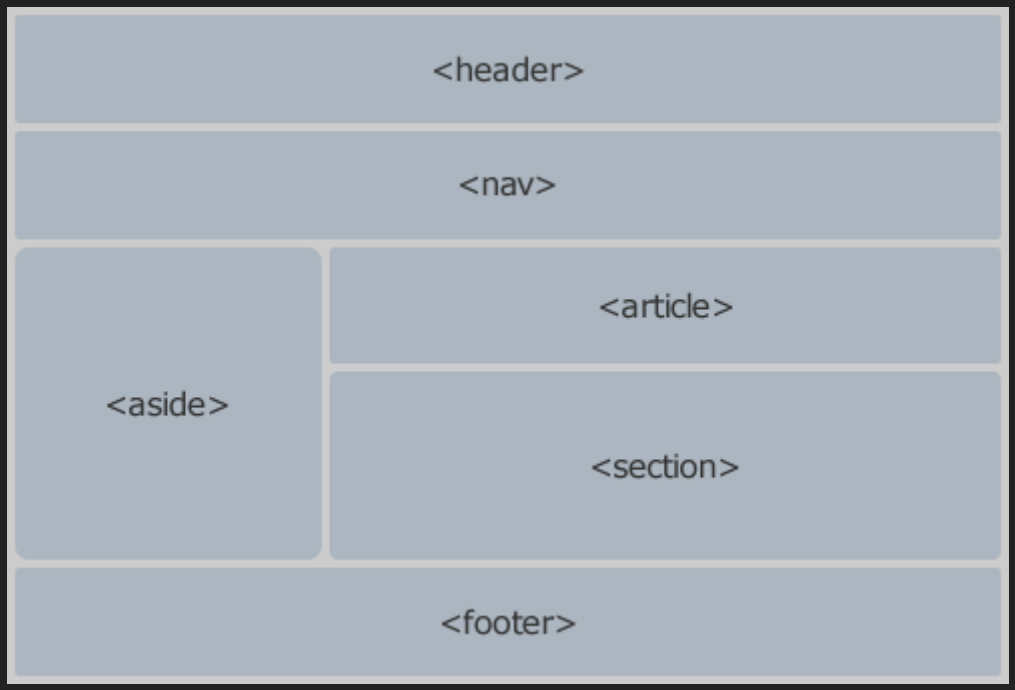
So, we'd wrap everything in DIVs and give it a class/id attribute to designate that portion of content.

Something like this:



With the advent of HTML5, we now have elements specific to the content within it.

Now, that same page might look like this...



The new HTML5 structural elements we'll go over:

- `<section>`
- `<header>`
- `<aside>`
- `<nav>`
- `<footer>`

SECTION

The `<section>` element represents a generic section of a document or application. A section, in this context, is a thematic grouping of content.

Similar to a `<div>`, but serves a more specific purpose

HEADER

The `<header>` element represents a container for introductory content or a set of navigational links.

FOOTER

The `<footer>` tag defines a footer for a document or section.

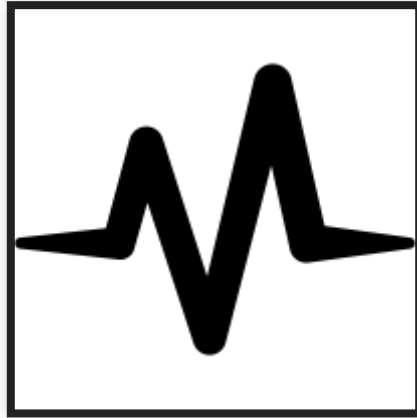
ASIDE

The `<aside>` element represents a portion of a document whose content is only indirectly related to the document's main content.

Asides are frequently presented as sidebars or call-out boxes.

NAV

The `<nav>` element represents a section of a page whose purpose is to provide navigation links, either within the current document or to other documents.



PULSE CHECK

Now that we've learned some of the more common structural elements in HTML5, lets see if you can draw an updated DOM including what we've learned today...

CLASS & ID

As you know, all HTML elements can have an **attribute**



And some HTML tags even require certain attributes:

```

```

There are some **attributes** that can be applied to *ANY* HTML element

These are referred to as **Global Attributes**, and today we're focusing on the 2 most common ones:

CLASS & ID

Classes and ids are special attributes that we can use to target specific elements on a page, so we can style/manipulate them uniquely.

The values for the `class` and `id` attributes can be whatever you want, but make sure its easy to understand what you're styling

GOOD:

```
<section class="blog-post-container">
```

BAD:

```
<section class="cookies">
```

**SO WHAT'S THE DIFFERENCE BETWEEN
THE CLASS AND ID ATTRIBUTE?**

ID

The `id` attribute is unique.

It can only be used once in an HTML document.

```
<div id="container">
```

```
...
```

```
</div>
```

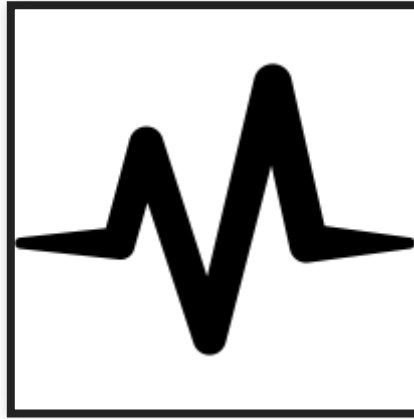
CLASS

The `class` attribute, on the other hand, is **NOT** unique.

It is designed to be used repeatedly in an HTML document.

```
<h1 class="center">Main Title</h1>
<div class="center">
    ...
</div>
```

So, when should you use them?



PULSE CHECK

You tell me...

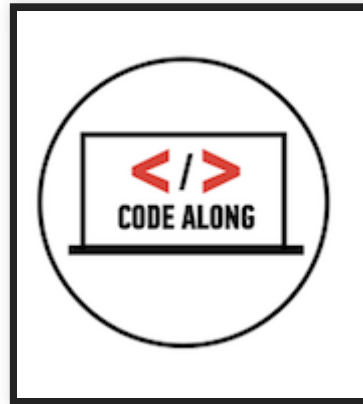
Now, how do we select these attributes in CSS?

ID

```
#container {  
  width: 1000px;  
  background: blue;  
}
```

CLASS

```
.center {  
  text-align: center;  
}
```



CLASS & ID CODE ALONG

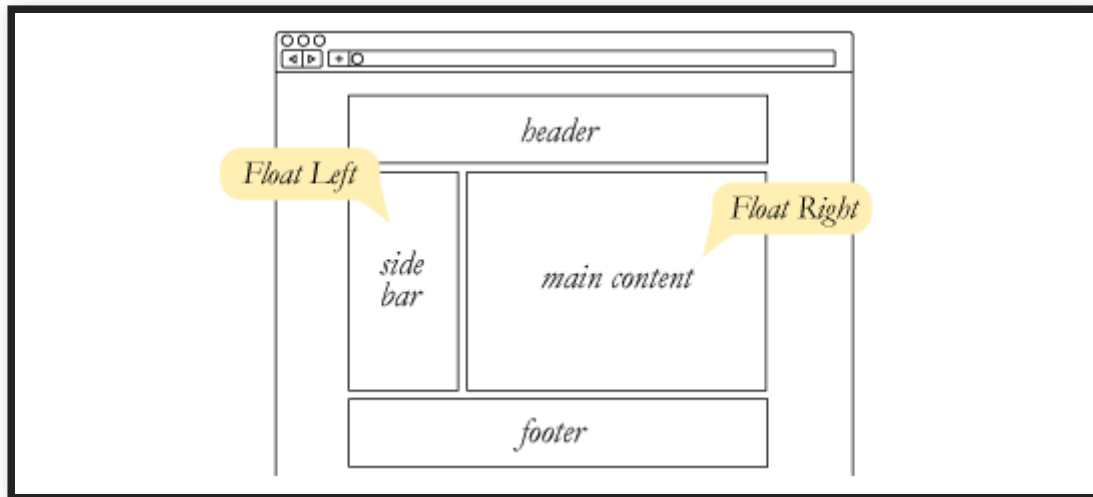
15 MINUTE BREAK



FLOATS

What is `float`?

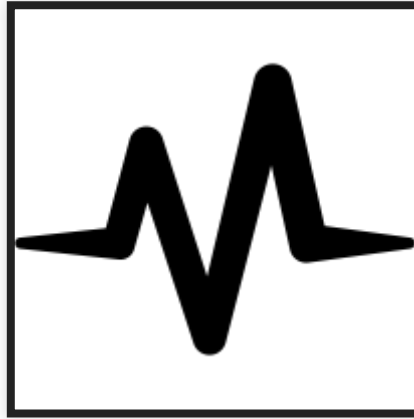
Simply put, `float` is a CSS positioning property, used to help layout a web page.



The `float` property looks like this in CSS:

```
.left {  
    float: left;  
/* This pulls the element to the left, allowing content to flow  
*/  
}
```

```
.right {  
    float: right;  
/* This pulls the element to the right, allowing content to flow  
*/  
}
```

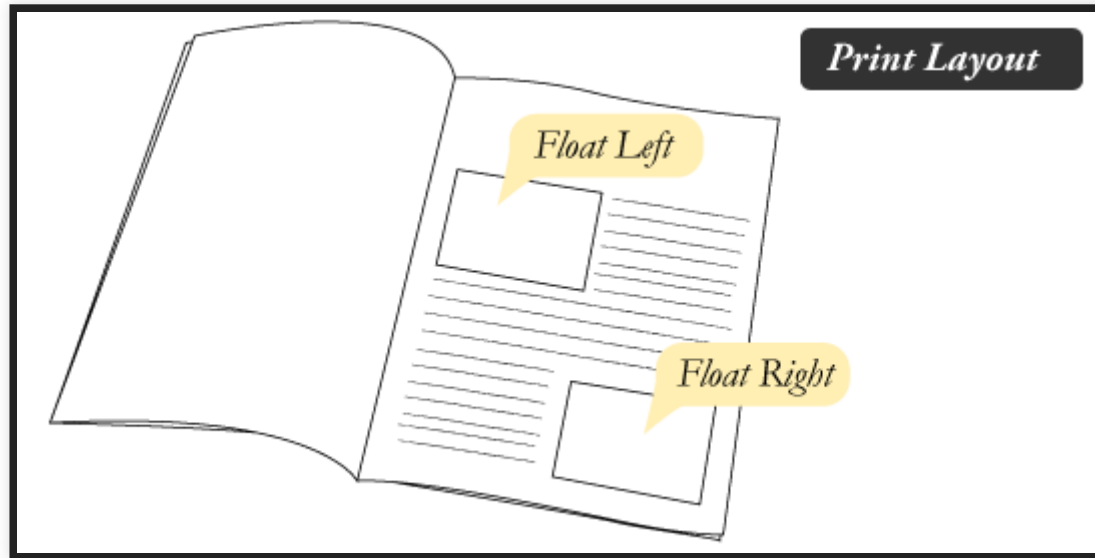


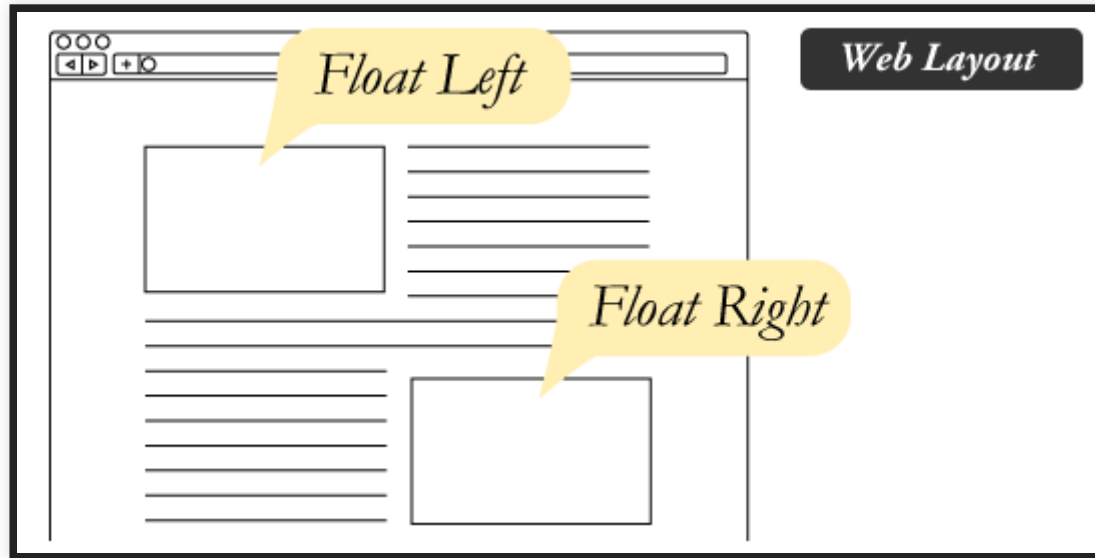
PULSE CHECK

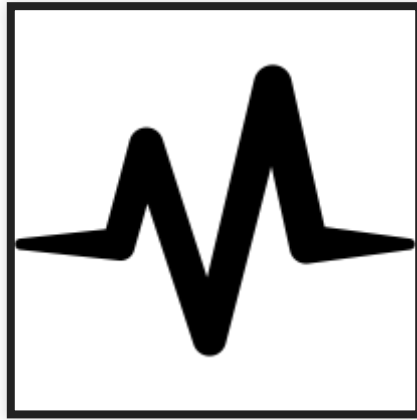
Let's add float to some boxes: [Div Float Codepen](#)

So, what does the `float` CSS property actually do?

The `float` property essentially allows content to flow around the floated element.







PULSE CHECK

Let's style a blog post, and I want you to decide which `float` property to use...

Image Float Codepen

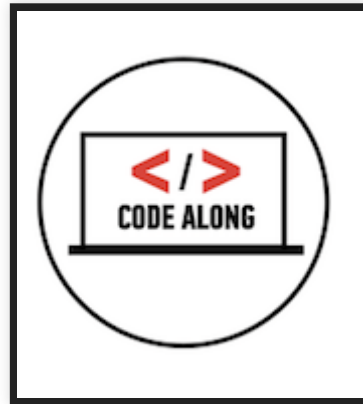
CLEARs

If you want an element to **NOT** wrap around a floated element, then you simply need to apply the `clear` CSS property to it

```
clear: both;
```

CSS FLOAT REFERENCES

- [CSS Tricks - All About Floats](#)
- [A List Apart - CSS Floats 101](#)
- Mozilla Developer Network
- W3Schools
- Codepen.io (Fork the Codepens from above and play around with the settings)



FLOATING SECTIONS



RELAXR LAB REVISIT

LEARNING OBJECTIVES REVIEW

- We Described and identified use-cases for normalize.css and reset.css files.
- We Drew an updated DOM Tree for web pages containing `<div>` & `<section>`
- We Applied header, footer, sidebar, and multi-column layouts to develop a web page.
- We Experimented and predicted effects of floats and clearing CSS positioning.

EXIT TICKETS

**NEXT CLASS: LESSON 05 -
LAYOUT LAB**

HOMEWORK

- Finish developing the single-column Relaxr page as outlined in the **Week 2/Assignment** folder
- The full design image can be found under **Week 2/Assignment/starter_code/images/relaxr_landing.jpg** and the content can be found in the design file under **Week 2/Assignment/starter_code/design_file.txt**
- The goal is to make sure your finished website looks as close to the design file as you can, while using markup we learned over the past 2 weeks
- Pay attention to requirements outlined in the rubric.m file