

WEEK 07
3/9/15

DIVs, Classes, and IDs

CORE LAB INTERACTION
MONDAY 7:00—9:40

This week

- Reading discussion
- Week 6 recap and questions
- DIVs, classes, and IDs
- Break
- Work session

Office Hours

BETTY

Mondays
5:00—6:30
back of the 6th floor,
6 East 16th St.

JACOB

By appointment.
I'm happy to help;
please email to
set up a time.

Last week's reading

- Who's Your Coding Buddy
- What's Network Neutrality?
- CAPS vs Mixed Case

CSS recap and questions

THIS WEEK

Giving your page structure & style

<div>

<div>

What are they?

The <div> tag defines a division or a section in an HTML document.

The <div> tag is used to group block-elements to format them with CSS.

<div>

A comparison



<div> are like clothespins:

- you always use two (<div> and </div>)
- one cannot work without the other
- they hold a block of code in place (like a piece of clothing)
- they control the structure, spacing, width, and layout of the object, but the content is undefined

<div>

Why would we need to do that?

We use <div> tags to group together HTML elements and apply CSS styles to many elements at once.

For instance, by wrapping a set of paragraph elements into a element, the developer can take advantage of CSS styles and apply a font to all paragraphs at once by applying a font style to the tag instead of coding the same style for each paragraph element.

<div>

What do they look like?

```
<div>  
  <!-- some grouped stuff -->  
</div>
```

<div>

When to use <div>

Where no existing HTML element is applicable, <div> can valuably represent parts of a document.

*wikipedia

Only use <div> for grouping objects.
Use the HTML elements we've already
talked about first.

THIS IS GOOD

<p>Hello, world!</p>

THIS IS NO GOOD

<div>Hello, world!</div>

THIS IS BEST

```
<div>  
  <p>Hello, world!</p>  
</div>
```


They look the same!

So why does it matter?

There are no right and wrong ways to add <div> tags to your page. Two different methods can produce the same results.

There are better and worse ways, though. Generally, the goal is to use as few as possible.

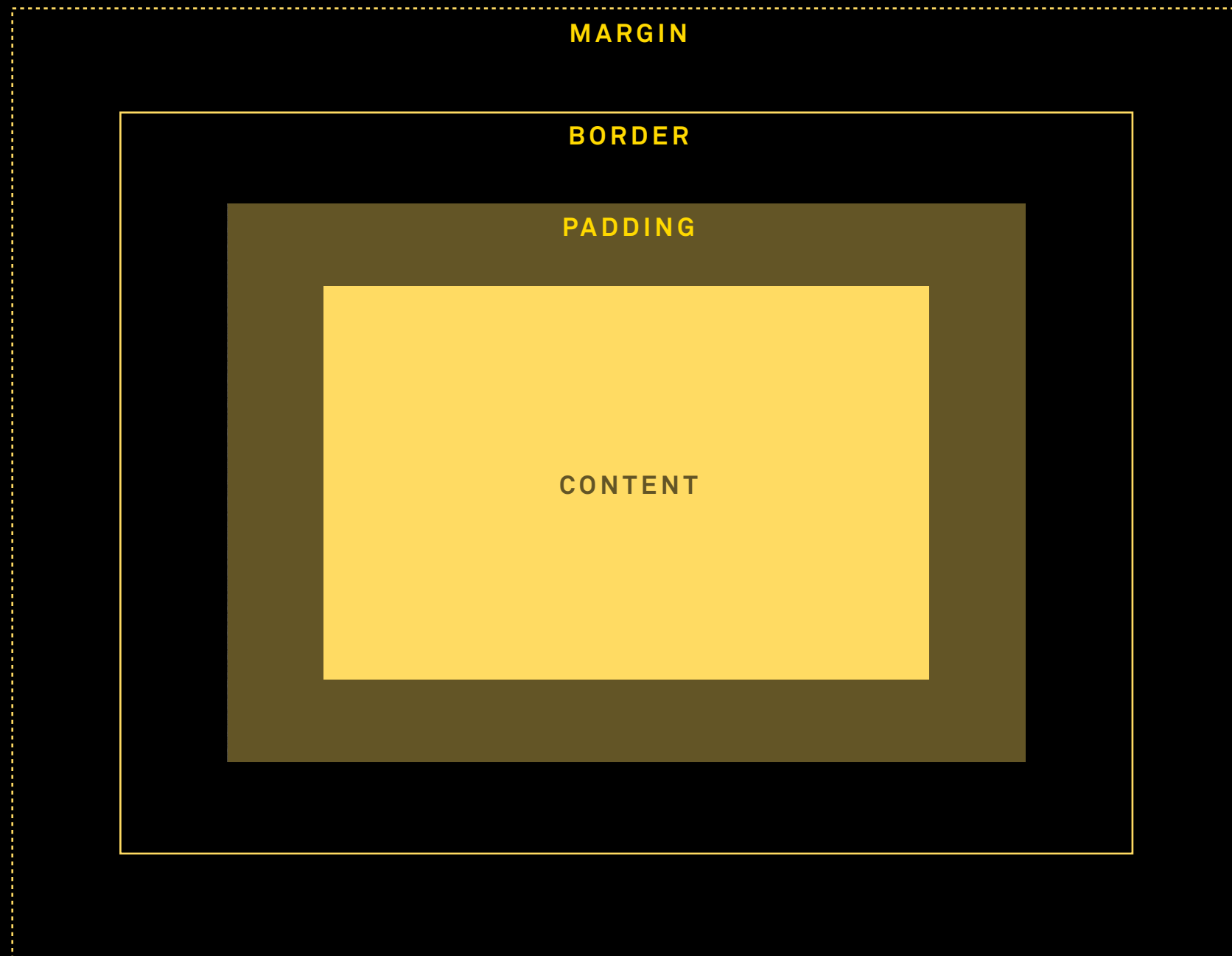
The Box Model

All HTML elements can be considered boxes. In CSS, the term "box model" is used when talking about design and layout.

The CSS box model is essentially a box that wraps around HTML elements, and it consists of: margins, borders, padding, and the actual content.

The box model allows us to add a border around elements, and to define space between elements.

The Box Model



Demo!

CSS selectors

Remember those guys?

```
selector {  
  property: value;  
}
```

CSS selectors

Remember those guys?

CSS selectors allow you to select and manipulate HTML elements.

CSS selectors are used to "find" (or select) HTML elements based on their id, class, type, attribute, and more.

We already did this last week

We just used HTML elements

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


This week
we're going to
use classes
and IDs

Classes and
IDs are how
CSS **targets**
specific HTML
elements

You can add
them to
anything so that
you can target
that element

We sometimes call these **hooks** in the code.



What's the difference?

CLASS

Classes are not unique. You can use the same class on multiple elements. You can use multiple classes on the same element

ID

IDs are unique. Each element can only have one ID, and each page can have only one element with that ID.

Let's use classes for describing how things should look, and IDs for identifying things.

Let's mostly use classes.

CLASS VS. ID

In HTML

CLASS

```
<div class="my-class">  
  <!-- stuff -->  
</div>
```

ID

```
<div id="some-id">  
  <!-- stuff -->  
</div>
```

CLASS VS. ID

In CSS

CLASS

```
.my-class {  
  color: red;  
}
```

ID

```
#some-id {  
  color: red;  
}
```


CLASS VS. ID

Together

CLASS

THIS GOES
IN YOUR
CSS

```
.my-class {  
  color: red;  
}
```

ID

```
#some-id {  
  color: red;  
}
```

THIS GOES
IN YOUR
HTML

```
<div class="my-class">  
  <!-- stuff -->  
</div>
```

```
<div id="some-id">  
  <!-- stuff -->  
</div>
```

Demo!

Naming classes

- Names should be semantic, meaning they should explain what they're doing
- Avoid reusing the same name for different things
- Names should follow consistent naming conventions
- use capitalization or hyphenation instead of spaces
- plan ahead so you can reuse classes

Demo!

Blocking out sections of HTML

There are different strategies for how to do this, but the basic goal is to “wrap” logically grouped elements.

Imagine moving a block without changing the layout or appearance of its contents.

Demo!

Adding DIVs, classes and IDs

Add DIVs, classes, and IDs

- Copy your Warby Parker wireframe page into a new project
- Add a CSS file and link to it in your HTML
- Wrap blocks of HTML into DIVs
- Add classes and styles for colors, type sizes, etc.
- Add a style that will give every block a red border

Break

NEXT WEEK

Sizing and positioning

Remember to turn in
the assignment and do the reading
before we meet