HTML & CSS: FUNDAMENTALS OF DEVELOPMENT

Instructor: Beck Johnson

Week 4

SESSION OVERVIEW

- Review CSS box model and inline/block
- Classes and IDs
- Layout – floats and positioning
- Using web fonts
- Supplemental technologies: Javascript, jQuery

REVIEW!

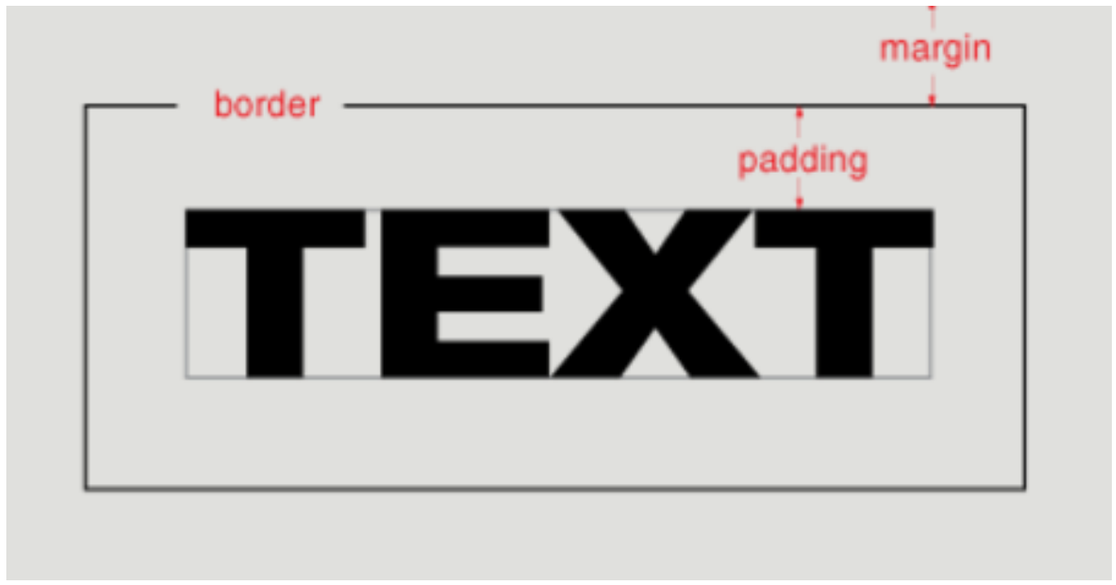
{ } CSS BOX MODEL

CONTENT: stuff in the box

PADDING: bubble wrap and packing peanuts

BORDER: sides of the box

MARGIN: space between multiple boxes



{ } PADDING

- Padding creates space **inside** an element
- Padding affects how far content is from the border

Padding is the space inside the border, between the border and the actual image or text.

{ } MARGIN

- Margin creates space **outside** an element.
- Margin affects how far elements are from each other

Margin is the space between one object and its surrounding elements.

{ } BORDER

Between margin and padding, you can set a `border`

- Width (usually in pixels)
- Border style (solid, dotted, dashed, etc)
- Color

```
p
    border: 2px dotted #ff0000;
}
```

{ } BORDER-RADIUS

To make an element appear curved, use the property `border-radius`

- The value is a number (in px or em) or percentage
- You can use `border-radius` even if you don't explicitly set a `border`

```
li {
      border-radius: 50%;
      height: 3em;
      width: 3em;
}
```

{ } BLOCK ELEMENTS

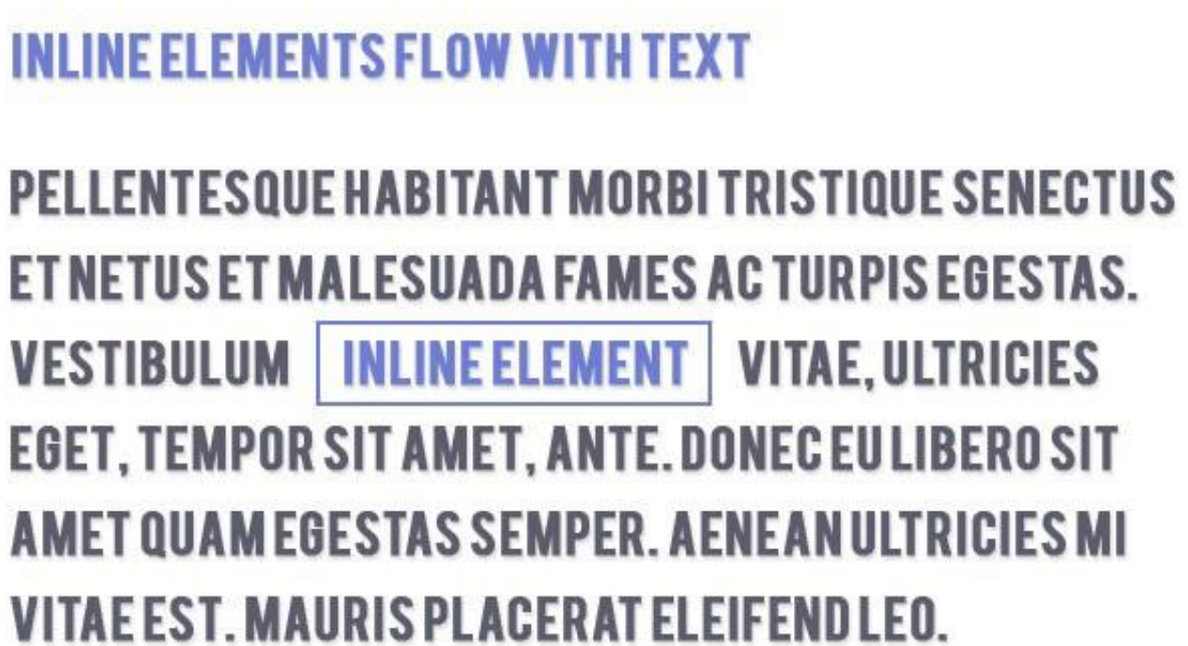
BLOCK ELEMENTS

- Expand naturally to fill their parent container
- Can have margin and/or padding

{ } INLINE ELEMENTS

INLINE ELEMENTS

- Flow along with text content
- Ignores height, width, top margin, and bottom margin
- Honors left and right margins (and any padding)



{ } INLINE-BLOCK ELEMENTS

INLINE-BLOCK ELEMENTS

- A hybrid of block and inline
- Flow along with text content
- Has height, width, margin, and padding

QUESTIONS?

ID & CLASS SELECTORS

CLASSES AND IDS

CSS lets us target **all** paragraphs like this:

```
p {
    font-size: 20px;
}
```

But what if we want to style only **some** paragraphs?

CLASSES AND IDS

You can add `class` and `id` attributes to any HTML element to identify it for styling.

- You decide the `class` and `id` names – be descriptive!

```
<p class="important">Big text</p>

<p class="anyLettersOrNumb3rsOr_Or-">Still
totally valid</p>
```

CLASSES AND IDS

Adding a `class` or `id` does nothing to an element by default.

- Classes and ids don't have any styling information by themselves
- They require you to add CSS if you want styling to be applied

CLASSES AND IDS

Multiple elements can have the same `class`

- A class is like a barcode – all of the same products have the same barcode

Only **one** element per page can use the same `id`

- An id is like a serial number – it uniquely identifies one specific instance of a product

CLASS SELECTORS IN CSS

- In CSS, target a `class` with a **period**
- Will style **all** types of elements that have that `class`:

```
.ghost { color: white; opacity: .1; }
```

```
<p class="ghost">Spooky!</p>

<div class="ghost">This will be spooky too.</div>
```

ID ATTRIBUTES

- An `id` can only be used **once** per page
- Elements **cannot** have multiple `id` attributes

```
<div id="mainContent">
     <!-- This better be the only main -->
</div>
```

ID SELECTORS IN CSS

```
<div id="lego"></div>
```

In CSS, target an id with a **hash**:

```
#lego {
    display: block;
}
```

IDS FOR ANCHORING

If you put a hash followed by the element's `id` in the URL, the browser will **jump** to that location on the same page:

```
<a href="#kittens">Proceed directly
to kittens</a>

...

<div id="kittens">Meow<div>
```

ID ATTRIBUTES

Q: What horrible thing will happen if you use an `id` twice on the same page?

A: Well…actually nothing.

- But your page won't validate
- Jump links will go to whatever `id` appears first
- And any Javascript that needs to locate that specific element will fail

HOW TO CHOOSE - CLASS OR ID?

If you think it's likely or possible that you'll want to apply the same style to multiple things, definitely use `class`

If your element is guaranteed to be the only one on the page, you can use `id` – or you can still use `class`

If your element needs to be linked to directly, use `id`

PRACTICE TIME!

ASSIGNMENT

Give an element on your page a descriptive `class`

- Apply a special style using a CSS `class` selector
- Style a **child** element of this element

Create another class and apply it to **two different** types of elements

- Bonus points: apply to an element that already has a class. What happens if the styles conflict? How would you make sure the result is what you want?

Assign an `id` to an element on your page

- Apply a unique style using an `id` selector
- Create a link in your nav that jumps to that element

WEB LAYOUTS

WEB LAYOUTS

With CSS, we can use a variety of properties to arrange elements on the screen by adjusting the flow of the page.

Basically, you can put elements anywhere…which can be both a good and a bad thing!

3 WEB LAYOUT PROPERTIES

- `display:` dictates how elements behave within the box model

- `float:` moves elements around within the page flow

- `position:` takes elements entirely out of the page flow

DISPLAY PROPERTY

The `display` property tells the browser what type of box model to use:

- `inline`
- `inline-block`
- `block`

This changes how padding, margin, height and width affect an element.

You also can set `display: none` to hide an element entirely.

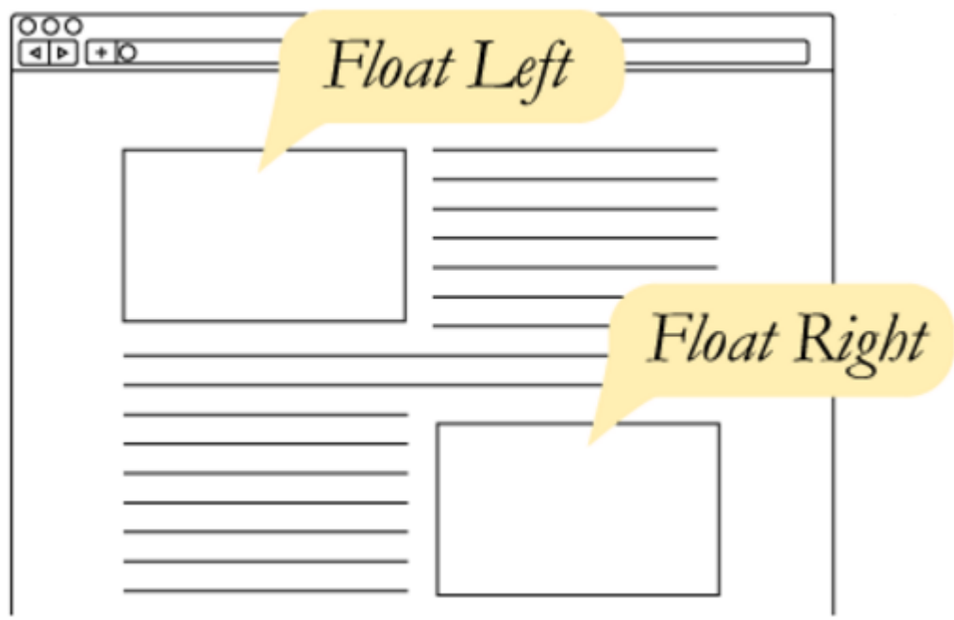
CSS FLOATS

Up to now, elements have displayed sequentially, in the order that you placed them in your HTML.

The `float` property takes an element out of the normal flow and "floats" it to the left or right side of its container.

- This allows other content to flow around it

```
img { float: left; }
```



CSS FLOATS

The three values for `float` are:

- `left`
- `right`
- `none`

By default, elements are `float: none`

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Explicabo sequi veniam ea enim nesciunt doloremque delectus sint consectetur qui magnam. Recusandae, hic quidem officia, asperiores sit libero sapiente totam eum.

Doloribus nisi ratione necessitatibus unde veritatis commodi veniam quas eaque fugiat nihil esse, id? Tempora quis quod impedit quia, facere incidunt, voluptatum dicta in dolores suscipit temporibus quam eos odit?

Doloribus nisi ratione necessitatibus unde veritatis commodi veniam quas eaque fugiat nihil esse, id? Tempora quis quod impedit quia, facere incidunt, voluptatum dicta in dolores suscipit temporibus quam eos odit?

CSS FLOATS

```
img { float: right; }
```

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Explicabo sequi veniam ea enim nesciunt doloremque delectus sint consectetur qui magnam. Recusandae, hic quidem officia, asperiores sit libero sapiente totam eum.

Doloribus nisi ratione necessitatibus unde veritatis commodi veniam quas eaque fugiat nihil esse, id? Tempora quis quod impedit quia, facere incidunt, voluptatum dicta in dolores suscipit temporibus quam eos odit?

Doloribus nisi ratione necessitatibus unde veritatis commodi veniam quas eaque fugiat nihil esse, id? Tempora quis quod impedit quia, facere incidunt, voluptatum dicta in dolores suscipit temporibus quam eos odit?

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Explicabo sequi veniam ea enim nesciunt doloremque delectus sint consectetur qui magnam. Recusandae, hic quidem officia, asperiores sit libero sapiente totam eum.

CSS FLOATS

```
img { float: left; }
```

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Explicabo sequi veniam ea enim nesciunt doloremque delectus sint consectetur qui magnam. Recusandae, hic quidem officia, asperiores sit libero sapiente totam eum.

Doloribus nisi ratione necessitatibus unde veritatis commodi veniam quas eaque fugiat nihil esse, id? Tempora quis quod impedit quia, facere incidunt, voluptatum dicta in dolores suscipit temporibus quam eos odit?

Doloribus nisi ratione necessitatibus unde veritatis commodi veniam quas eaque fugiat nihil esse, id? Tempora quis quod impedit quia, facere incidunt, voluptatum dicta in dolores suscipit temporibus quam eos odit?

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Explicabo sequi veniam ea enim nesciunt doloremque delectus sint consectetur qui magnam. Recusandae, hic quidem officia, asperiores sit libero sapiente totam eum.

FLOWS BEFORE ROWS

Where you place your floated element in your HTML can cause different results.

Live demo here.

```
<div id="container">
   <img src="rootbeer-float.png" style="float:right" />
   <p>This is some text contained…</p>
</div>
```

FLOWS BEFORE ROWS

What happens if we move the image below the paragraph?

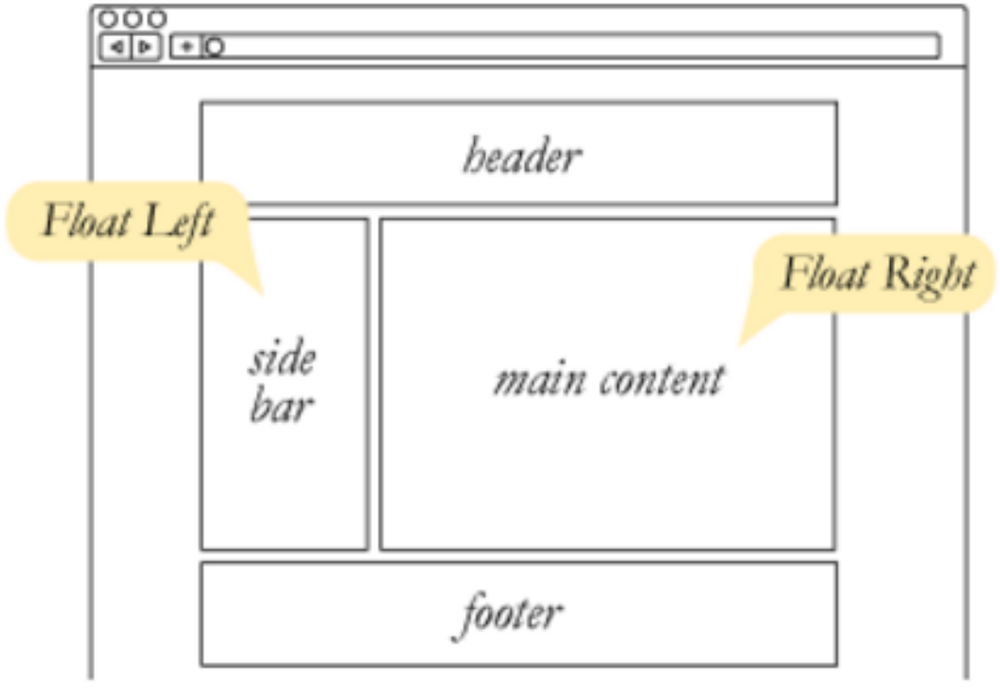
```
<div id="container">
   <p>This is some text contained…</p>
   <img src="rootbeer-float.png" style="float:right" />
</div>
```

A general rule is **float first**.

- Place floated elements before any non-floated elements the float interacts with

CSS FLOATS

`float` can be used to float text around images, but it also can be used to create entire page layouts.



CSS FLOATS

For example, this layout was built using `float`.
How do you think it was done?

BY BECK JOHNSON

Beck Johnson is a developer based in Seattle, WA. She plays board games in her spare time. Say hi at **beckjohnson@gmail.com**

```
<div>
   <img class="avatar" src="beck.png" />
   <h2 class="bio">By Beck Johnson</h2>
   <p class="bio">Beck Johnson is… </p>
</div>
```

CSS FLOATS

BY BECK JOHNSON

Beck Johnson is a developer based in Seattle, WA. She plays board games in her spare time. Say hi at **beckjohnson@gmail.com**

```
.avatar { float: left; }
```

```
.bio { float: right; }
```

Let's try that...

CSS FLOATS

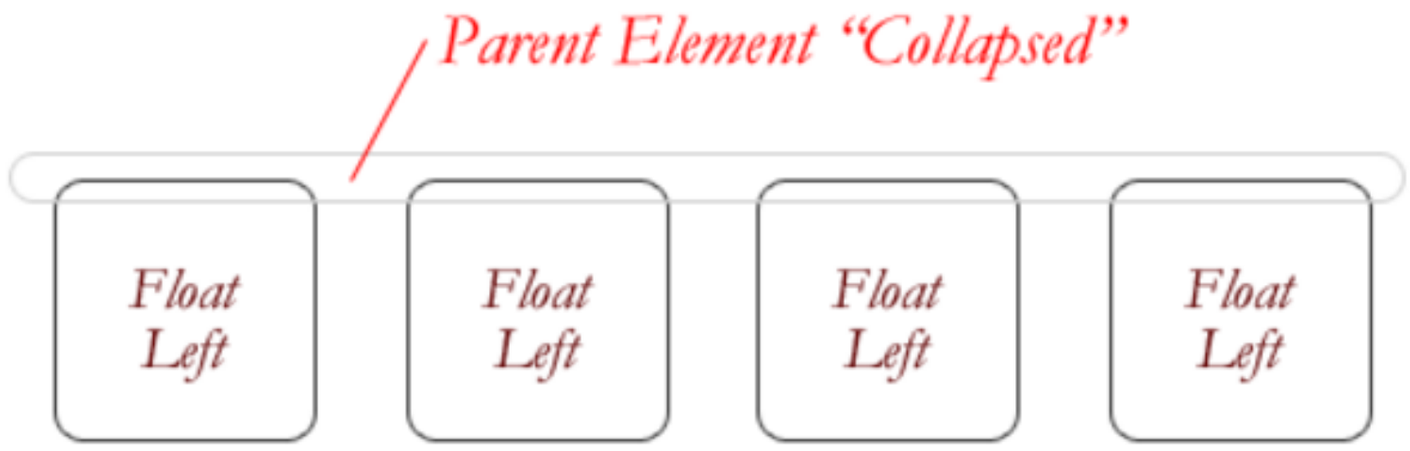
BY BECK JOHNSON

Beck Johnson is a developer based in Seattle, WA. She plays board games in her spare time. Say hi at **beckjohnson@gmail.com**

The container thinks it has no content!

- It collapsed to the size of its padding (you can see the top and bottom `border`)
- The floated content is spilling out

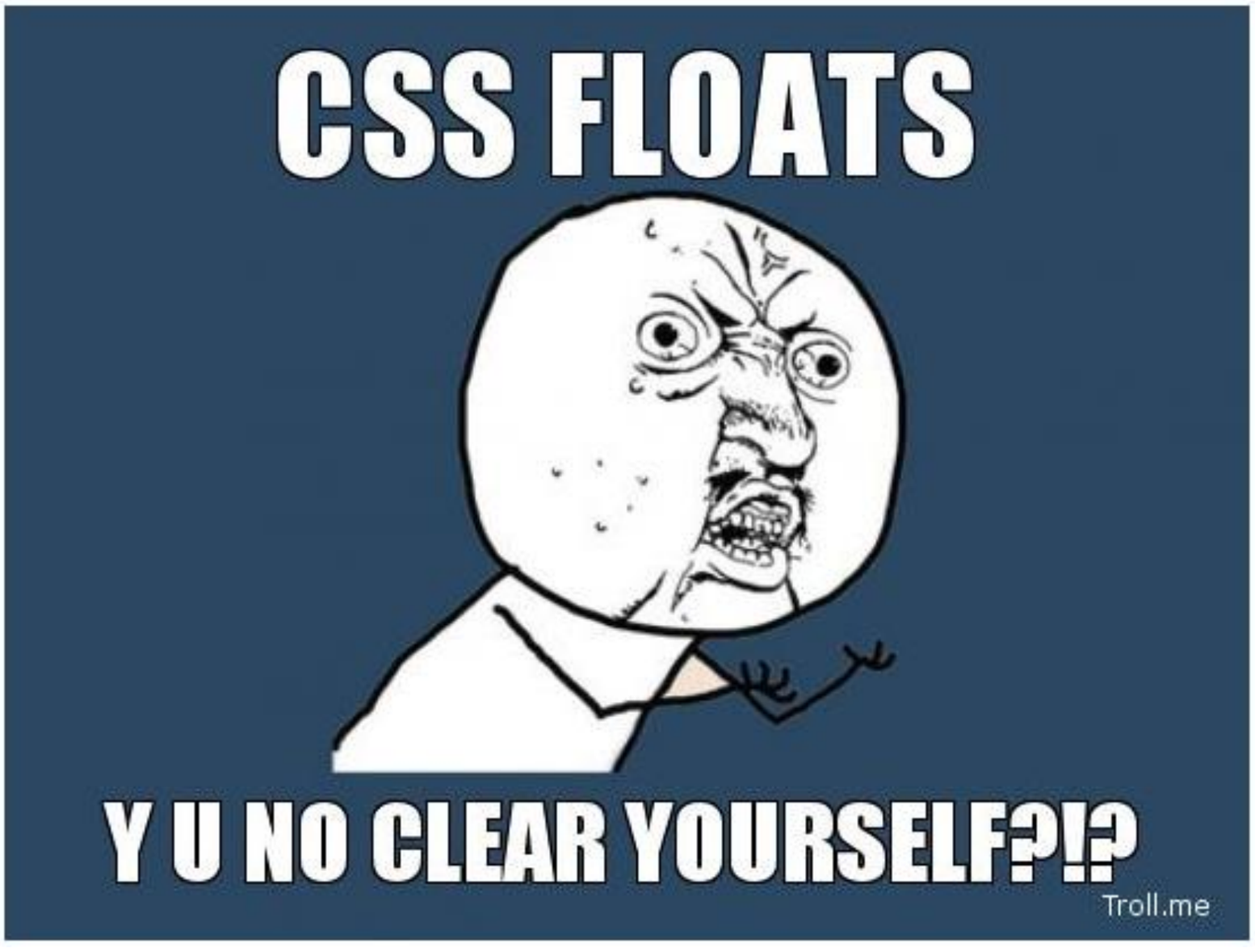
CSS FLOATS



If you float an element, it is no longer in the normal document flow.

If all elements in a container are floated, that means that the container is effectively "empty."

HOW TO FIX FLOATS?



HOW TO FIX FLOATS?

There are 2 ways to fix this:

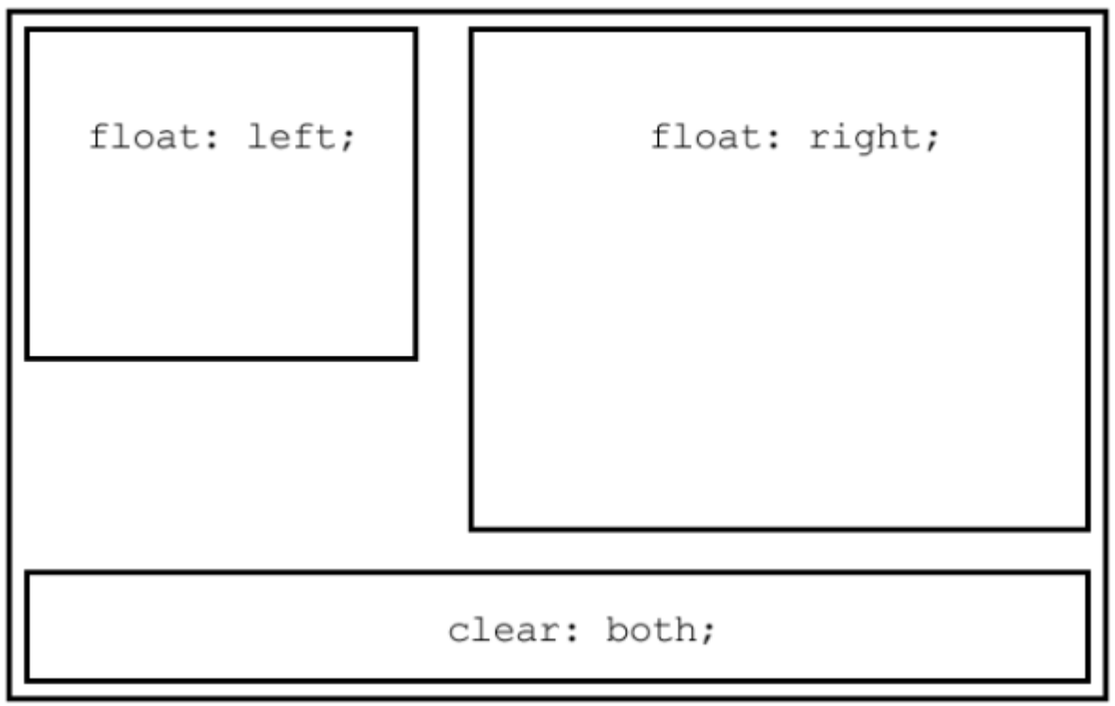
1. Apply the CSS rule `clear: both` to an element **after** the floated content

2. Apply a CSS rule using the property `overflow` to the **container**

THE CLEAR PROPERTY

The `clear` property is the sister property to `float`

- It doesn't do much until there are floated elements on the page
- An element with `clear` applied to it will force itself **below** the floated element
- Everything after that will be back in the normal flow
- This "stretches" out the container and keeps it from collapsing

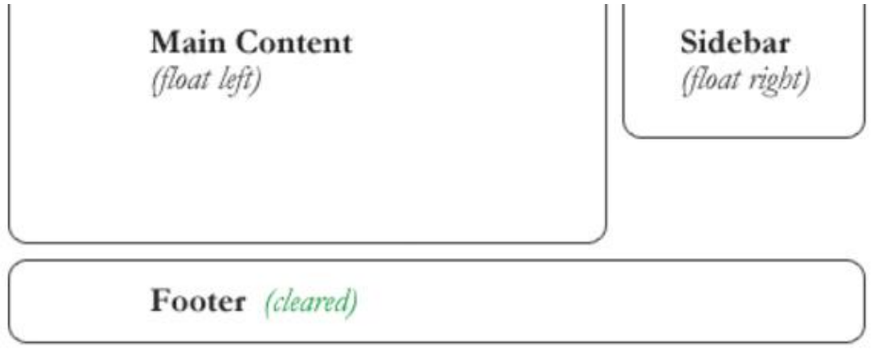


THE CLEAR PROPERTY

`clear` has similar values to `float`:

- `clear: none` – the element does **not** move down to clear past floating elements (this is the default value)

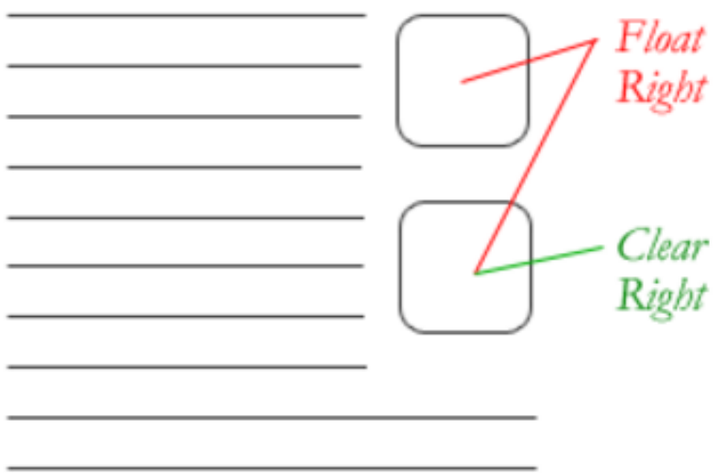
- `clear: both` – the element is moved down to clear **both** right- and left-floated elements



THE CLEAR PROPERTY

Sometimes, you want to let some content after a `clear` continue floating, but not other content:

- `clear: left` – only clear **left-floated** elements
- `clear: right` – only clear **right-floated** elements



THE CLEAR PROPERTY

So to solve our problem, you could add this empty `div` after the bio container:

BY BECK JOHNSON

Beck Johnson is a developer based in Seattle, WA. She plays board games in her spare time. Say hi at **beckjohnson@gmail.com**

```
<div style="clear: both"></div>
```

(We could apply the class to any type of element, but the benefit of using a div is that it has no style of its own.)

THE OVERFLOW PROPERTY

The other way to force a container to expand around floated content is to apply a CSS rule with `overflow` to the container that the floated content is inside.

Any valid value for `overflow` will cause floated content to stretch out the container

- Too complicated to explain, but it basically forces the container to re-assess the content inside it

THE OVERFLOW PROPERTY

`overflow` is a CSS property that governs how content looks when it breaks out of its container.

By default, elements have `overflow: visible`, which means all content is fully visible

- Even if that means overflowing its container!

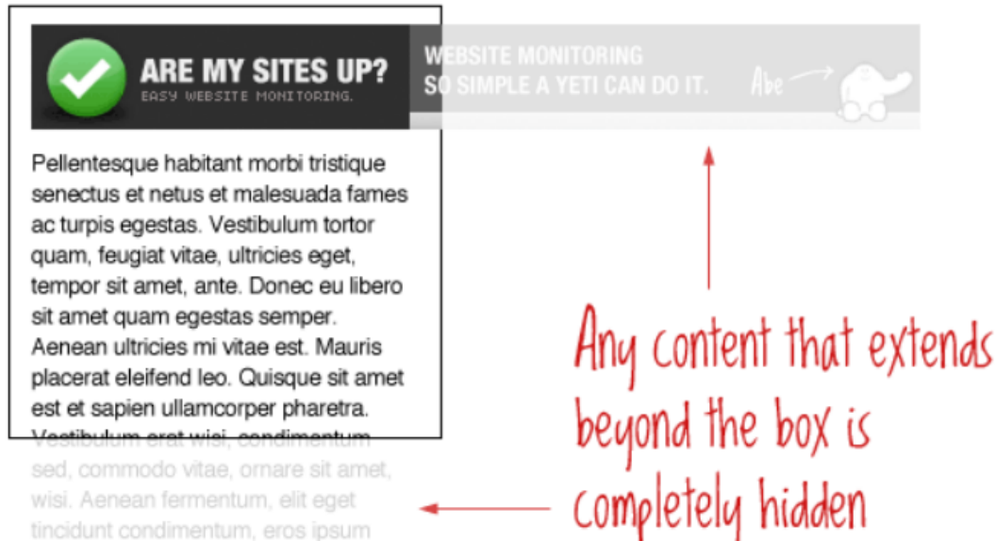
THE OVERFLOW PROPERTY

`overflow: scroll` makes scrollbars appear both horizontally and vertically…even if they don't need to be there.

- None of the content that would overflow appears outside the box

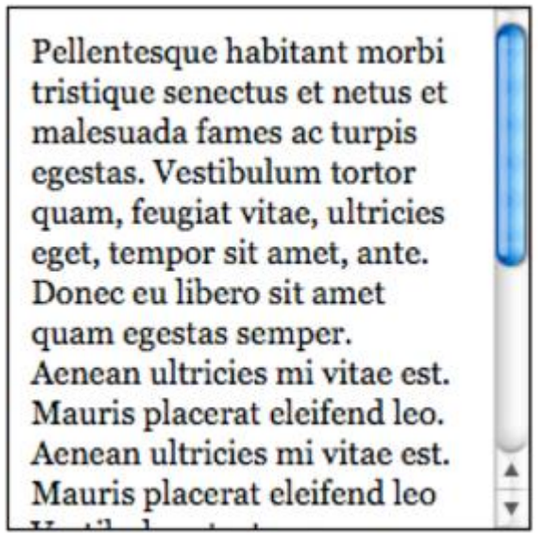
`overflow: hidden` cuts off any content that "sticks out" of its box

- No way to scroll, so content is no longer accessible



THE OVERFLOW PROPERTY

`overflow: auto` only adds scrollbars when the content requires it (which may mean no scrollbars are added at all)



THE OVERFLOW PROPERTY

So to solve our problem, you could add this CSS rule to the floated `div`:

```
.bio { overflow: visible; }
```

PRACTICE TIME!

ASSIGNMENT

Create a container that has an image floated to the side of some text.

- Give the container a background color, gradient, or borders (to make sure floated content is really clearing properly)

- Try both techniques to force the container to expand around floated content

- Apply box model properties like padding and margin so that your content looks nice

WEB FONTS

WEB FONTS

Remember that `font-family` looks for a font installed on the user's local machine.

```
body { font-family: Tahoma, sans-serif; }
```

If the Tahoma font isn't found, the browser will default to a generic sans-serif font instead.

What if you want to use an interesting font that most people aren't likely to have installed?

WEB FONTS

The absolutely easiest way to get custom fonts is to link to a CDN font stylesheet in the head of your page:

```
<link href="https://fonts.googleapis.com/css?family=Roboto"
rel="stylesheet">
```

Remember that CDN means “Content Delivery Network”

- A fancy way of saying a big, reliable company hosts the file for you
- Google hosts 100s of free fonts

WEB FONTS

```
<link href="https://fonts.googleapis.com/css?family=Roboto"
rel="stylesheet">
```

Once you include a font stylesheet from a CDN, you can refer to the font in a `font-family` rule, just like you would a web-safe font:

```
p { font-family: Roboto, sans-serif; }
```

WEB FONTS

1 Family Selected

Your Selection Clear All

Roboto

EMBED CUSTOMIZE Load Time Fast

Embed Font

To embed your selected fonts into a webpage, copy this code into the <head> of your HTML document.

STANDARD @IMPORT

```
<link href="https://fonts.googleapis.com/css?family=Roboto" rel="styleshe
et">
```

Specify in CSS

Use the following CSS rules to specify these families:

```
font-family: 'Roboto', sans-serif;
```

When you choose a font Google will tell you what name to use.

WEB FONTS

Roboto
Christian Robertson (12 styles)

All their equipment and instruments are alive.

Baloo Bhaina
Ek Type (1 style)

A red flair silhouetted the jagged edge of a wing.

Try typing directly into the text fields. GOT IT

Diplomata SC
Eduardo Tunni (1 style)

I WATCHED THE STORM, SO BEAUTIFUL YET TERRIFIC.

Open Sans
Steve Matteson (10 styles)

Almost before we knew it, we had left the ground.

Lato
Łukasz Dziedzic (10 styles)

A shining crescent far beneath the flying vessel.

Slabo 27px
John Hudson (1 style)

It was going to be a lonely trip back.

Ranga
TipTopTyp (2 styles)

Mist enveloped the ship three hours out from port.

Oswald
Vernon Adams, Kalapi Gajjar, Cyreal (6 styles)

My two natures had memory in common.

Roboto Condensed
Christian Robertson (6 styles)

Silver mist suffused the deck of the ship.

Source Sans Pro
Paul D. Hunt (12 styles)

The face of the moon was in shadow.

Montserrat
Julieta Ulanovsky (18 styles)

She stared through the window at the stars.

Griffy
Neapolitan (1 style)

The recorded voice scratched in the speaker.

https://fonts.google.com/

WEB FONTS

CONS:

- Relies on a 3rd party to provide assets
- If the request times out, the font won't download (so always provide fallbacks!)

PROS:

- Extremely easy to get working
- Possibility that user already has the font in their web cache due to visiting other sites that use the same font

ICON FONTS

Font Awesome is a free icon font that is used in many real-world projects.

bullhorn	bullseye	bus	cab (alias)
calculator	calendar	calendar-check-o	calendar-minus-o
calendar-o	calendar-plus-o	calendar-times-o	camera
camera-retro	car	caret-square-o-down	caret-square-o-left
caret-square-o-right	caret-square-o-up	cart-arrow-down	cart-plus
cc	certificate	check	check-circle
check-circle-o	check-square	check-square-o	child
circle	circle-o	circle-o-notch	circle-thin

An icon font means letters have been replaced with vector images

- So to color or re-size icons on your site, just use the CSS `font` properties we already learned.

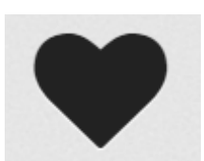
ICON FONTS

To quickly start using Font Awesome, add this stylesheet to the head of your page:

```
<link href="https://maxcdn.bootstrapcdn.com/font-awesome/4.7.0/css/font-awesome.min.css" rel="stylesheet">
```

Click an icon from the Font Awesome website, and copy the markup they provide, like:

```
<i class="fa fa-heart"></i>
```



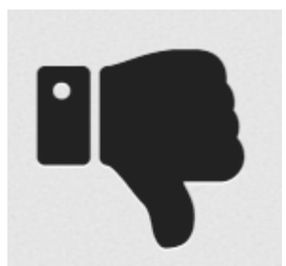
FONT AWESOME

You can put a Font Awesome class on any element:

```
<span class="fa fa-paw"></span>
```

```
<i class="fa fa-cog"></i>
```

```
<h1 class="fa fa-thumbs-down"></h1>
```



WEB FONTS

The other way to ensure people see the correct font is to download the font onto their computer when they load your page.

- You do this by first downloading a font into a folder on your website

- There are free fonts available for download at websites like Font Squirrel or Font Spring

- After you have a copy of the font you can share it just like any other media file on your website

WEB FONTS

Some considerations with hosting fonts:

- Have to consider licensing fees – many fonts cost money
- Some fonts prohibit commercial use, or limit the number of pageviews

@FONT-FACE

If you download a font that you want to use on your site, add `@font-face` before any other styles:

```
@font-face {
  font-family: MyWebFont;
  src: url('webfont.eot');
       url('webfont.eot?#iefix') format('embedded-opentype'),
       url('webfont.woff') format('woff'),
       url('webfont.ttf') format('truetype'),
       url('webfont.svg#svgFontName') format('svg');
}
```

(The good news is, font sites will provide this chunk of CSS for you to copy into your stylesheet!)

@FONT-FACE

```
@font-face {
  font-family: MyWebFont;
  src: url('webfont.eot');
       url('webfont.eot?#iefix') format('embedded-opentype'),
       url('webfont.woff') format('woff'),
       url('webfont.ttf') format('truetype'),
       url('webfont.svg#svgFontName') format('svg');
}
```

Different browsers support different font filetypes

- Modern browsers use woff
- IE needs eot
- Mobile devices need ttf or svg

PRACTICE TIME!

ASSIGNMENT

Find a free font from Google fonts and use it on your site by including the font stylesheet.

- Apply the font to some (or all) elements on the page

Include the Font Awesome stylesheet:

```
<link href="https://maxcdn.bootstrapcdn.com/font-awesome/4.7.0/css/font-awesome.min.css" rel="stylesheet">
```

- Display at least two different icons
- Make them different sizes and/or colors
- Bonus points: what other CSS can you apply to the icons?

OVERVIEW OF JAVASCRIPT

WHAT IS JAVASCRIPT?

JavaScript is a programming language that runs in your web browser

- Can manipulate any element on the page
- Can listen to user interaction, such as button clicks or scrolling the browser window

WHERE TO PUT JAVASCRIPT?

JavaScript goes inside a new tag called `<script>`

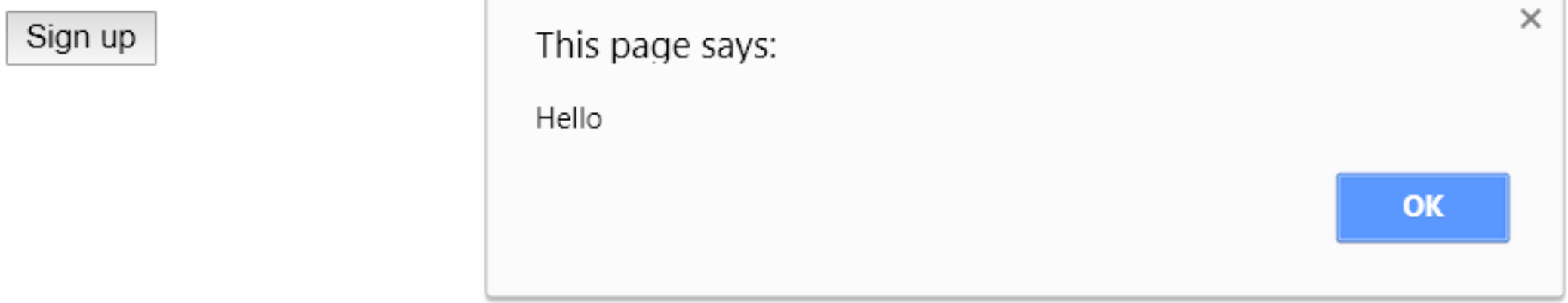
- Just like how anything inside a `<style>` tag is expected to be a different "language" than your HTML document (CSS), everything inside a `<script>` tag is expected to be in the Javascript language

- Unlike malformed CSS, you'll see red errors in the console if the browser can't understand how to read things inside `<script>`

JAVASCRIPT EXAMPLE

```
<button id="signup">Sign up</button>

<script>
   var signupButton = document.getElementById("signup");
   signupButton.addEventListener("click", function(){
      alert("Hello");
   });
</script>
```



See a live demo

WHAT IS JQUERY?

jQuery is a popular Javascript library that makes it easy to locate and manipulate elements

- Uses CSS syntax to locate DOM elements
- Handles browser inconsistencies (so that you don't have to)
- Simplifies many common tasks

USING JQUERY

To be able to use jQuery, you have to include a link to the Javascript file that contains the jQuery library

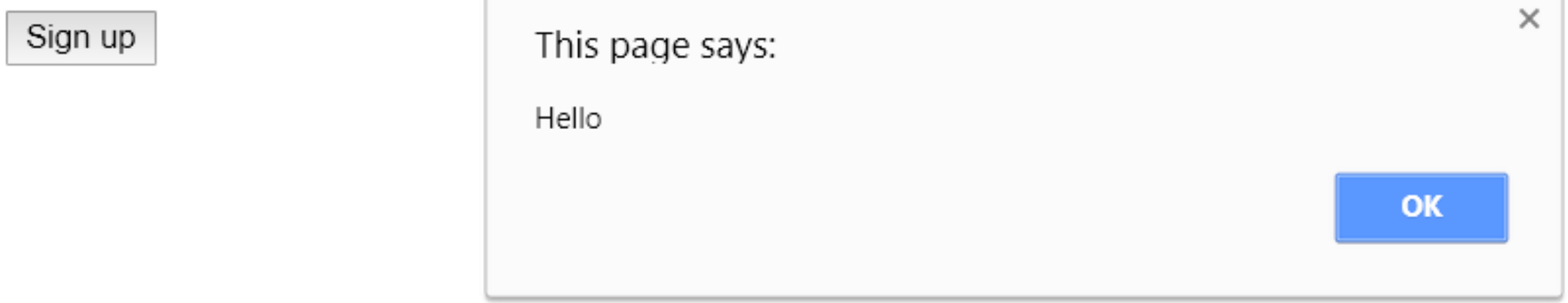
- Like web fonts, the easiest way to get started is to link to a CDN:

```
<script src="https://code.jquery.com/jquery-
1.12.4.js"></script>
```

JQUERY EXAMPLE

```
<button id="signup">Sign up</button>


<script>
   var signupButton = $("#signup");
   signupButton.click(function(){
      alert("Hello");
   });
</script>
```



BETTER JQUERY EXAMPLE

```
<a class="menu-toggle">Top level menu</a>
<ul class="menu">
     <li>Menu item</li>
     <li>Menu item</li>
     <li>Menu item</li>
</ul>

<script>
   $(".menu-toggle").click(function(){
          $(".menu").slideUp();
          $(this).next(".menu").slideDown();
   });
</script>
```

See a live demo

JQUERY PROVIDES EASY ANIMATIONS

jQuery makes showing or hiding elements very easy, and provides a bunch of animation options as well

- Show/hide
- Fade in / fade out
- Slide down / slide up

Each of these options can be given a duration (how long the animation lasts) and a speed (called “easing”, which can vary over time)

See some jQuery animation examples

“HOMEWORK”

- Practice!
- Optional: read chapters 15 and 17 of *HTML and CSS: Design and Build Websites*