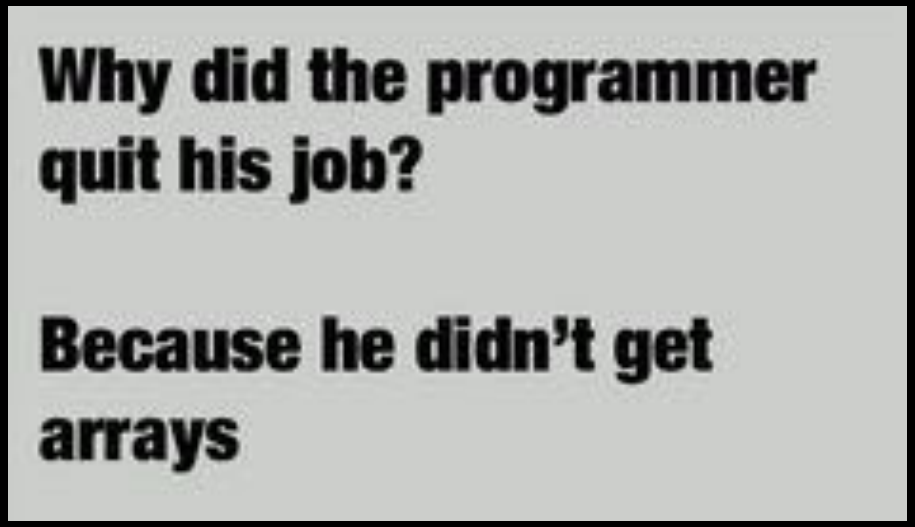
WEEK 7 – JQUERY, CSS POSITIONING

GENERAL ASSEMBLY

FEWD



Joe Bliss
Dad-Joke Enthusiast

AGENDA

Review Homework
Continue with jQuery
CSS Positioning
Codealongs

ANIMATIONS - .ANIMATE();

```
$("#myElement").animate({
        "opacity": ".3",
        "width": "500px",
        "height": "700px"
    }, 2000);
```

The .animate() method allows us to create animation effects on (most) numeric CSS properties.

.ANIMATE() THESE THINGS:

"border-width"
"border-bottom-width"
"border-left-width"
"border-right-width"
"border-top-width"
"margin"
"margin-bottom"
"margin-left"
"margin-right"
"margin-top"
"padding"
"padding-bottom"
"padding-left"
"padding-right"
"padding-top"
"height"
"width"
"font-size"
"top"
"left"
"bottom"
“right"

CALLBACK FUNCTION

A callback function is a function that runs AFTER an effect is completed. It is super useful for timing things around animations.

For example:

```
$("p").fadeOut(2000 ,function(){
    alert("The paragraphs are now gone.");
});
// This will fadeOut all the paragraphs on the page over 2 seconds, then, once that has completed, send an alert.
```

CHAINING COMMANDS

Chain chain chain ... Chain of fools ...

Perform multiple effects one after the other. Only works when called on the SAME object.

```
$("#p1").fadeIn(800).slideUp(2000).slideDown(2000);
```

CHAINING COMMANDS

You can chain any jQuery functions, not just Effects. When you chain Effects, they are queued.

When you chain normal functions (i.e. not Effects), they execute immediately one after another.

.delay(ms) will delay Effects by that number of milliseconds (and have no effect on non-Effects).

EXERCISE - JQUERY EXERCISE

Include jQuery via Google CDN, or downloading it and linking it locally.

Create your own Javascript file.

When the user mouses over a box:
- Change the color, inner text.
- Perform some animations on the box - .animate()

When the user clicks on a box:
- Hide that box with some effect. - .hide(), .fadeOut(), .slideUp()
- After that effect has run, .remove() that box from the DOM.

JQUERY

~~DOM Manipulation~~

~~Effects / Animation~~

Document Traversal

Events

DOCUMENT TRAVERSAL

How we "get-around" within our document. How each element is related to all the other elements on the page.

http://api.jquery.com/category/traversing/

IT'S ALL IN THE FAMILY

.children()

- Get the child elements

.parent()

- Get the parent element

.siblings()

- Get the sibling elements

BROTHERS AND SISTERS

.next()

- Get the immediately following sibling element

.next("some selector")

- Get the immediately following sibling element but only if it matches "some selector”

.prev()

- Get the immediately preceding sibling element

BROTHERS AND SISTERS

.index()
Returns an item's index within its siblings.

.eq(someNumber)
Returns the item at the specified index (someNumber)

CODEALONG - CAROUSEL

We want to make this generic for ANY number of slides and any navigation? How can we do this?

Think about the DOM tree structure ...

JQUERY

~~DOM Manipulation~~

~~Effects / Animation~~

~~Document Traversal~~

Events

EVENTS

"These methods are used to register behaviors to take effect when the user interacts with the browser, and to further manipulate those registered behaviors."

We've already looked-at .click();

http://api.jquery.com/category/events/

MOUSE EVENTS

```
$("#somediv").mouseenter(function() {
  //fired when the mouse enters an element
});

$("#somediv").mouseleave(function() {
  //fired when the mouse leaves an element
});

$("#somediv").hover(function() {}, function() {});
// hover()is shorthand for mouseenter() and mouseleave() above
```

KEYBOARD EVENTS

```
$(“#somediv”).keypress(function() {
  //fired when the keyboard is pressed on an element
});
```

WINDOW EVENTS

```
$(window).resize(function() {
  //fired when browser is resized
});

$(window).scroll(function() {
  //fired when browser is scrolled
});
```

CODEALONG – MORE ROBUST LIST MAKER

A more robust List Maker.

- Delete an entry.
- Do some validation.
- Accept “enter” key submission.

CODEALONGS

CSS POSITIONING

Static
Fixed
Relative
Absolute

What's your favorite posish?

POSITION: STATIC;

Everything that we've seen so far has been "position: static;" by default. This is the default for all elements.

You cannot set right, left, top, bottom values to elements with position: static;

A static element is said to be "not positioned" and an element with its position set to anything else is said to be "positioned".

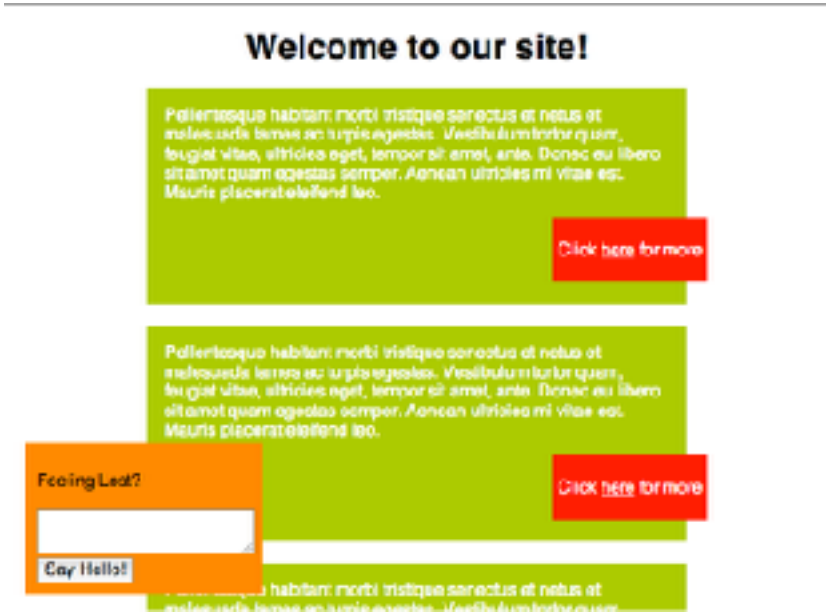
POSITION: FIXED;

Fixed-position elements don't move when the browser scrolls. They are placed relative to the browser window (top, left, bottom, right). Elements with position: fixed; are removed from the normal flow of the page.

It can be a bit "buggy" in older browsers, as well as unsupported in some mobile browsers. (http://caniuse.com/css-fixed)

Usage: Persistent navigation, "Modal" divs, Animation

POSITION: FIXED;



Create a chat window:
https://codepen.io/ga-joe/pen/BZWyzB

POSITION: ABSOLUTE;

An absolutely-positioned element is positioned with respect to it's nearest parent with a position other than static (relative, absolute, fixed). By default, it is positioned with respect to the browser window.

Absolutely-position elements are taken out of the normal flow of the page.

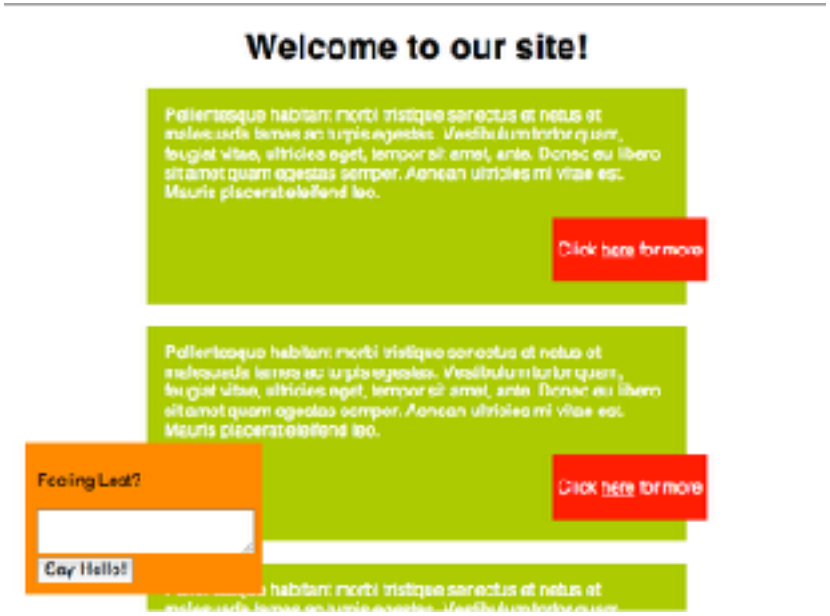
Usage: Animation, Placing an element within a parent.

POSITION: ABSOLUTE; + POSITION: RELATIVE;

We can use position: relative; together with position: absolute; to place an object WITHIN another object.

You place position: relative; on the parent object, and then, children of that object with position: absolute; applied to them will determine their placement on the position: relative; object.

POSITION: ABSOLUTE; + POSITION: RELATIVE;



https://codepen.io/ga-joe/pen/BZWyzB

CODEALONG - SLIDESHOW

EASING

How can we approximate more natural movement in our apps? It's not difficult. It's EASING!

Easing functions specify the speed at which an animation progresses at different points within the animation.

Right now, our animate() function can only accept two types of easing - "linear" and "swing"

EASING

```
$(this).animate({
        top: "80%"
}, 2000, "swing");
```

- This is the default

```
$(this).animate({
        top: "80%"
}, 2000, "linear");
```

- This is built-in.

JQUERY EASING PLUGIN

Download or hotline from:
https://cdnjs.com/libraries/jquery-easing

For a visual cheat-sheet for all the different options:
http://easings.net/

CODEALONG - EASING

Change the way the boxes "fall".

EXERCISE - PACMAN ANIMATION

Place a couple characters on the screen and move them around using easing.

FINAL PROJECT MILESTONE

First draft of one page / section of HTML / CSS and at least one JS pseudocode interaction.

This can be extremely rough, just want to make sure you are working on it.