UNIT 2!

Part 1: Front End Javascript

INTRO

- We've spent a lot of time so far getting to know javascript on an intimate level. The true foundation to making usable tools with javascript.
- Now we're going to transition into building things!
- As we talked about in day 1, JS's original and primary application is to aid us with front end development.
- Let's talk more about that.

JAVASCRIPT IN THE FRONT END

- Do you guys remember what we had said about compiling languages?
- What is special about the way that JS compiles?
- You're right (probably, maybe)! JS is unique because it is the only true programming language that is compiled in the user's browser.
- This is a needed feature, let's chat about that in a minute.

JAVASCRIPT IN THE FRONT END

- As you know, javascript is one of the three major languages of the front end.
- HTML: responsible for the structure of a website
- CSS: responsible for the styling of the website
- JS: responsible for interaction with the website and getting/sending data from a website to and from servers.

JAVASCRIPT IN THE FRONT END

- Today we are going to focus on the interactive side of JS.
- What do we mean by interaction?
- When we talk about the interactive element of a website we mean making our website respond to user actions.
- For example:
  - Handling visual effects when a user hovers over something.
  - Changing the active slide in a slideshow.
  - Executing an action when a user clicks a button
  - Handling data when a user submits a form.
  - Anytime a user does something with the mouse or keyboard, JS is there to listen and respond.

SO WHAT DOES IT TAKE FOR JAVASCRIPT TO HANDLE INTERACTIONS?

- Let's back this up a bit. When a user clicks on something they are clicking on some kind of html element.
- This means that javascript needs to be able to communicate with html. At the very least (truly, the least) it needs to be able to tell what was clicked on, hovered on, etc.
- This means that ***javascript is connected to html.*** Mind you that this is not automatic, we need to declare which JS file is linked to that html page.
- Once we have the JS file linked, the JS needs code that finds which html elements it should pay attention to. More on this in a minute.

SO WHAT DOES IT TAKE FOR JAVASCRIPT TO HANDLE INTERACTIONS?

- We also know that sometimes we hover on things on a site, and a color might change. What does this tell us?
  - *JS can add interactivity (or "events") to html elements*
  - *JS can change css properties of html elements*
- We have also seen things like popups, or nav bars that appear out of nowhere or after a user action. What does this tell us?
  - *JS can create html elements and apply them to the page*
- Over the course of the next couple days we'll be going over strategies as how to do that.

THE DOM

- The struggle is that HTML and JS are two completely different languages, so how do they communicate?
- This is all handled in the browser. The browser creates something called the Document Object Model.
- ***The DOM is an object-based representation of the HTML document.*** It is essentially an attempt to convert the structure and content of the HTML document into an object model that can be used by various programs.
- It is used by browsers as a first step towards determining what to render in the viewport, and by Javascript programs to modify the content, structure, or styling of the page.

THE DOM TREE

- The DOM keeps track of what elements are on the page (even the hidden ones) and which elements have other elements inside of them.
- Each element is considered a “node” in the DOM. When visualizing this, the result looks a lot like a tree diagram. Hence, it is often referred to as a DOM tree.
- This makes it easy to keep track of where elements are relative to each other.

THE DOM TREE

- Terms:
  - An html element that is outside of another element is called “the parent element” of that element.
  - The inner element would be referred to as the “child element” of the outer element.
    - Elements that share the same parent are called “sibling elements”.
- The concept of a DOM tree becomes a bit more concrete with actually seeing the diagram. Let’s make on together on the whiteboard of this website:

THE DOM TREE

daocloud

FOR PROFESSIONALS

SIGN IN

USING THE DOM

- Three slides ago I said, ***The DOM is an object-based representation of the HTML document.*** So, the DOM is a (potential) large object that describes the structure of our content. Since it's an object, we can use normal techniques to get and set data! In the browser, the DOM is represented by the `document` object. JS specifies some built-in methods that make using the DOM easier.
- Let's look at some of the methods, or functions that belong to the document object

```
Document.getElementById('profile-pic');
Document.querySelector('#profile-pic');
Document.querySelectorAll('.messages');
```

CODEALONG

No better way to learn it then to do it

(there is a markdown for the codealong)

JQUERY

- jQuery is a cross-browser JavaScript library designed to simplify the client-side scripting of HTML.
- “Cross browser” - works the same in all* browsers.

- Allows:
  - Document traversal
  - CSS Manipulation
  - Event Handling
  - Animation
  - and more!

WHAT IS A LIBRARY?

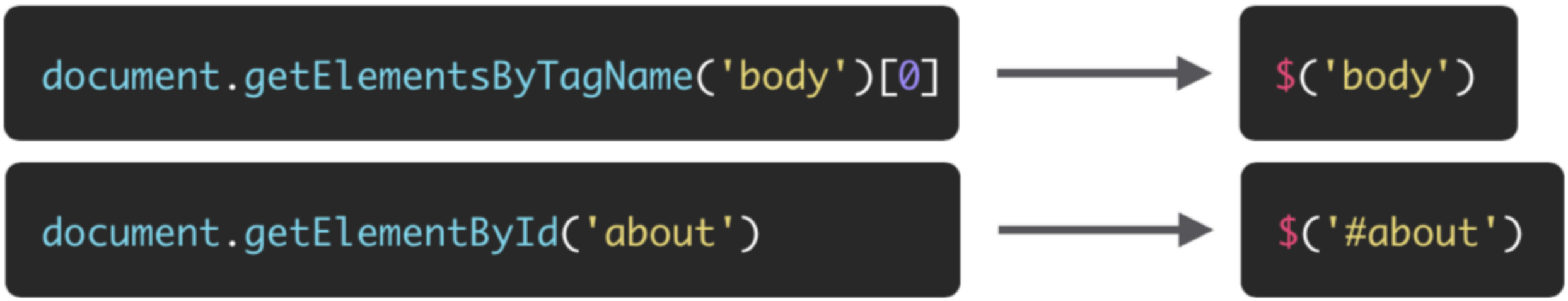
- Let's go to the jQ source code and take a look.
- A library is a tool for developers. It is a bunch of code written by someone else to solve common problems for developers.
- See how the source code of jQuery is just javascript? Thank goodness we didn't have to write all that!

INCLUDING JQUERY

- Adding jQuery to your website
- Download and link to the file
  - <script src="jquery-3.3.1.min.js"></script>
- Link to a remote version (CDN)
  - <script src="https://code.jquery.com/jquery-3.3.1.js"></script>
- A CDN is a “content delivery network”. It is a way to host raw code on the internet so that other apps can reference and include that code in their files.

JQUERY VS DOM SYNTAX

- there are formal rules around how to write it. This is the syntax.



HOW TO USE JQUERY

- To use it, just type in the $() to invoke the jQuery function.
- The $ is actually just a placeholder for “jQuery()”. You can also type jQuery(‘.images’) and it works as well. For obvious reasons the $ is more popular.
- After the $() we put a string inside that is a CSS selector (much like how ‘querySelector` and `querySelectorAll` works).

```
$(‘p’) // selects all p tags

$(‘.images’) // selects all elements with class “images”

$(‘#profile-pic’) // selects element with id “profile-pic”
```

HOW TO USE JQUERY

- jQuery also allows us to use any CSS selectors we want to query the DOM. For example this is totally valid:

```
$(“.specialList li:first-child")
```

- It definitely requires some CSS ninja abilities to leverage that type of selection, but it is possible!

HOW TO USE JQUERY

- Selecting something with jQuery is different then selecting things with “vanilla” JS.
- jQuery selections will always return a massive object called the jQuery object.
- The jQuery object has loads of information about the selected element and much much more.

THE JQUERY OBJECT

- The jQuery object comes loaded with a bunch of functions that we can use to easily change properties of an HTML element, update attributes, add events and more.
- To use a jQuery method (function), just type `.` with the method name afterwards. For example, let's look at one of the simpler ones:

```
var $item = $('#item');

$item.css('color', 'red');
```

- Once you've selected the item, you can just use the css method with .css('css-property', 'css-value')

THE JQUERY OBJECT

- All jQuery methods work in a similar fashion.
- Being successful with jQuery requires some encyclopedic knowledge of the functions it offers as well as the arguments it needs to work.
- Let's go over a few of them.

SLIPS

-Find a partner
- research a method
- present it

GROUPS:

- Group 1: .addClass(), removeClass(), toggleClass()
- Group 2: .append(), .prepend()
- Group 3: .html(), .text()
- Group 4: .show(), .hide(), .toggle()
- Group 5: .attr(), .val()
- Group 6: .fadeIn(), fadeOut(), fadeToggle()
- Group 7: .slideDown(), slideUp(), slideToggle()

- For your presentation give a clear definition of what it is used for. Describe what is being manipulated in the HTML/CSS. Write a codeine that demonstrates it in use.