



EVENTS & JQUERY

Code 301

Welcome to class!

Today we will look into the very HEART of JavaScript, and see how to unlock it's secrets with jQuery.

AGENDA

- Assignment review
- DOM Events
- jQuery Events
- jQuery Effects

ASSIGNMENT REVIEW

How'd things go with your first assignment? 👍👎

Let's do some code review. What brave soul will welcome improvement, inspection?

Let's talk through what we see here.

DOM EVENTS

Remember DOM Events? That little magic trick you can do in the browser to let the user know they clicked on something?

Turns out it's pretty important.

DOM EVENTS

- When _____, do _____.
 - When the user does something, run some code.
 - When an event is fired, invoke a function.
 - We register a callback function as an event handler
-
- It's a simple concept...with some radical implications
 - Out of order code execution
 - aka: "Asynchronous Callbacks"

Fill in the blanks for me!

This is how we talk about it. Note the usage of the vocab words!

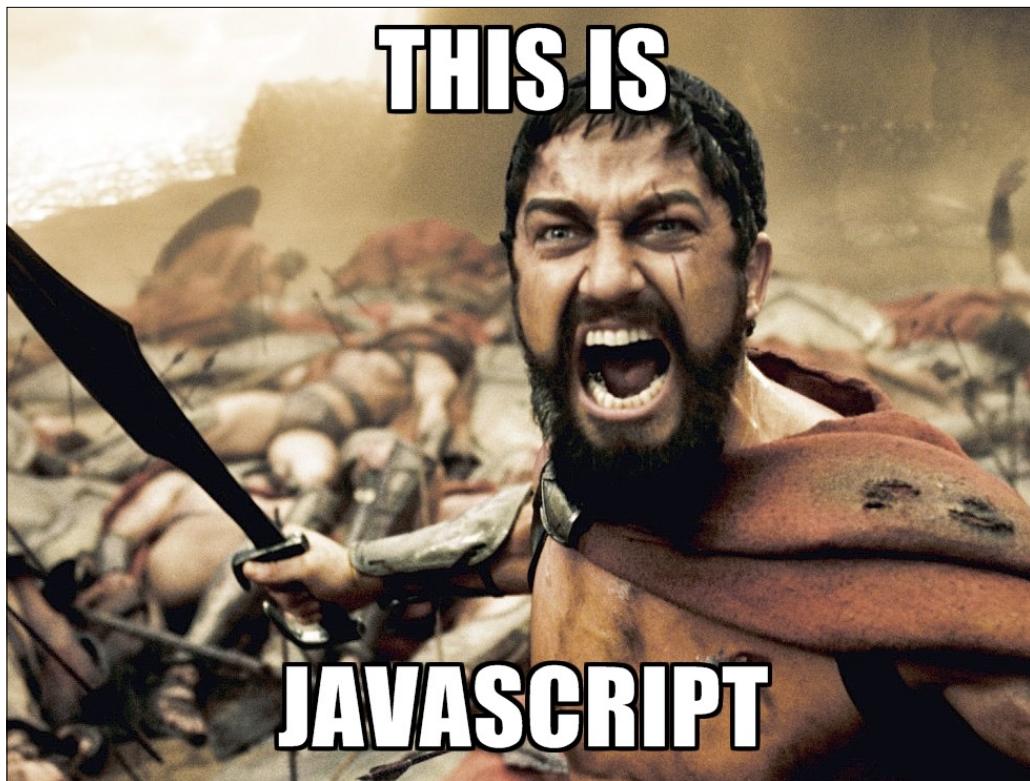
A beautifully simple concept: run some code when something happens.

Your file is no longer procedural, it's not top to bottom any more. Interpreter may not execute line 42 after line 41.

Ok with this?

Getting how this works?

Hope so, because...



...because: THIS IS JAVASCRIPT!

DOM EVENTS: THERE ARE LOTS OF THEM...

<code>checking</code>	<code>Event</code>	Offline	The user agent is checking for an update, or attempting to download the cache manifest for the first time.
<code>click</code>	<code>MouseEvent</code>	DOM L3	A pointing device button has been pressed and released on an
<code>close</code>	<code>Event</code>	WebSocket	A WebSocket connection has been closed.
<code>complete</code>		IndexedDB	
<code>complete</code>	<code>OfflineAudioCompletionEvent</code>	Web Audio API The definition of 'OfflineAudioCompletionEvent' in that specification.	The rendering of an OfflineAudioContext is terminated.
<code>compositionend</code>	<code>CompositionEvent</code>	DOM L3	The composition of a passage of text has been completed or canceled.
<code>compositionstart</code>	<code>CompositionEvent</code>	DOM L3	The composition of a passage of text is prepared (similar to keydown for a keyboard input, but works with other inputs such as speech recognition).
<code>compositionupdate</code>	<code>CompositionEvent</code>	DOM L3	A character is added to a passage of text being composed.
<code>contextmenu</code>	<code>MouseEvent</code>	HTML5	The right button of the mouse is clicked (before the context menu is displayed).
<code>copy</code>	<code>ClipboardEvent</code>	Clipboard	The text selection has been added to the clipboard.
<code>cut</code>	<code>ClipboardEvent</code>	Clipboard	The text selection has been removed from the document and added to the clipboard.
<code>dblclick</code>	<code>MouseEvent</code>	DOM L3	A pointing device button is clicked twice on an element.
<code>devicelight</code>	<code>DeviceLightEvent</code>	Ambient Light Events	Fresh data is available from a light sensor.
<code>devicemotion</code>	<code>DeviceMotionEvent</code>	Device Orientation Events	Fresh data is available from a motion sensor.
<code>deviceorientation</code>	<code>DeviceOrientationEvent</code>	Device Orientation Events	Fresh data is available from an orientation sensor.
<code>deviceproximity</code>	<code>DeviceProximityEvent</code>	Proximity Events	Fresh data is available from a proximity sensor (indicates an approximated distance between the device and a nearby object).
<code>dischargingtimechange</code>	<code>Event</code>	Battery status	The dischargingTime attribute has been updated.

MDN lists 193 “Standard Events”, and dozens more that are in development, or browser specific: <https://developer.mozilla.org/en-US/docs/Web/Events>

DOM EVENTS: THE ONES WE MOST CARE ABOUT

- A selection of useful events (p 246-247):
 - Browser UI: `load`, `unload`, `resize`, `scroll`
 - Mouse: `click`, `dblclick`, `mouseover`, `mouseout`
 - Keyboard: `keyup`, `keydown`, `keypress`
 - Element focus: `focus`, `blur`
 - Form: `input`, `change`, `submit`

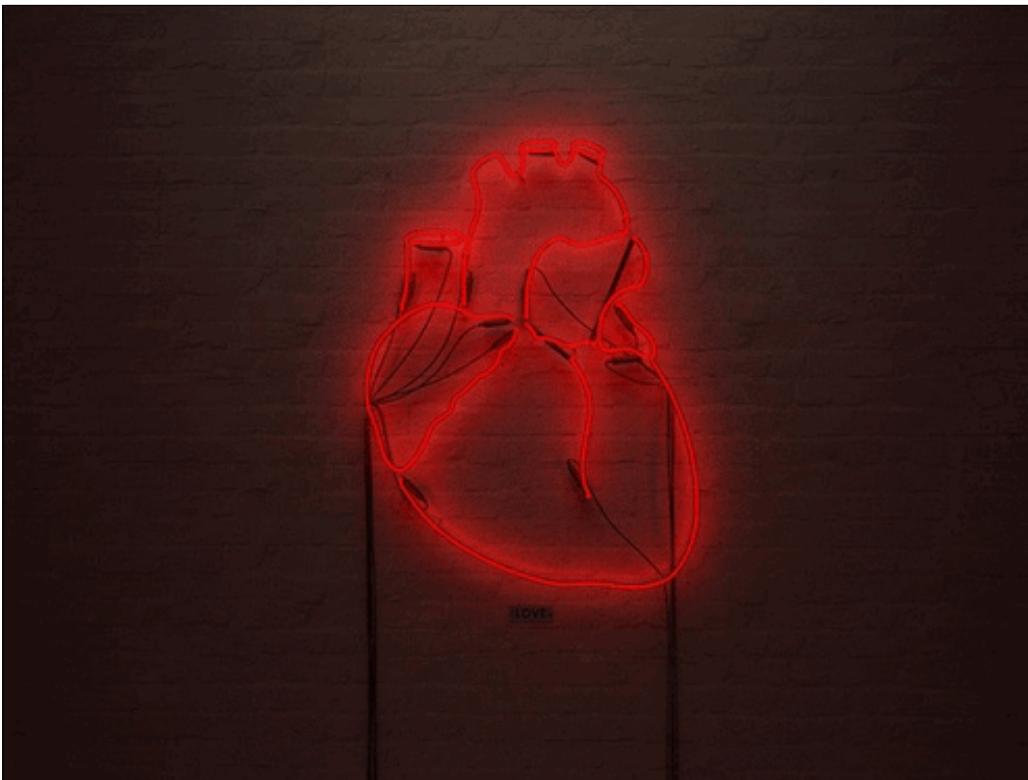
But there is a small handful that are particularly useful.

“

Events are the beating heart of any
JavaScript application.

-Peter-Paul Koch, quirksmode.org

Why do you think he says this?



Why do you think he says this?

EVENTS WITH JQUERY

- Browser event handling, jQuery style!

- Goodbye:

```
<a href="#" onclick="jumpHigh();">Jump!</a>  
domEle.onclick = function() { height = prompt('How high?'); };  
domEle.addEventListener('click', jumpHigh, true);
```

- Hello:

```
$el.on('click', jumpHigh);
```

- Let's take a look: <http://api.jquery.com/on/>

Identify the callback function in each line of vanilla js code.

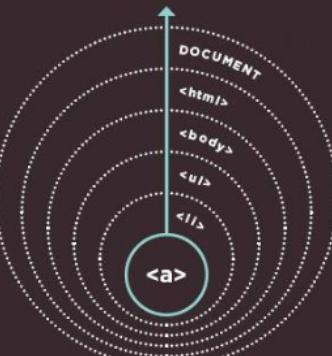
The question to ask yourself, whenever you see a callback, is: When will this function actually be invoked? When will this code run?

In each case, an event is *registered* when the interpreter executes the line of code, but the function specified doesn't run, until the event fires.

Let's read through the documentation here so you can see what to pay attention to on these docs pages.

HTML elements nest inside other elements. If you hover or click on a link, you will also be hovering or clicking on its parent elements.

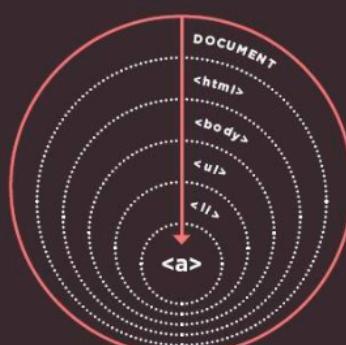
Imagine a list item contains a link. When you hover over the link or click on it, JavaScript can trigger events on the `<a>` element, and also any elements the `<a>` element sits inside.



EVENT BUBBLING

The event starts at the *most specific node* and flows **outwards** to the *least specific one*. This is the default type of event flow with very wide browser support.

Event handlers/listeners can be bound to the containing ``, ``, `<body>`, and `<html>` elements, plus the `document` object, and the `window` object. The order in which the events fire is known as **event flow**, and events flow in two directions.



EVENT CAPTURING

The event starts at the *least specific node* and flows **inwards** to the *most specific one*. This is not supported in Internet Explorer 8 and earlier.

Remember this? Page 260. We'll demo this in a minute. But recall your event propagates from one element through all parents.

EVENTS WITH JQUERY

- What events should our blog respond to?
 - ‘As a reader, I want to click the "Read On" button so that I can see the entire article.’
 - ‘As an author, I want my articles filtered by name so that readers can find all my posts.’
 - ‘As the creator, I want the About nav to act as a tab, so my story is revealed FAST.’
- DEMO: Your Blog, as it will be after today’s user stories are implemented

Let's see what these look like in action. We'll focus on the end result here, not the implementation just yet.

EVENTS DEMO

Let's play around with a few examples.

Sample code: p 329, 331

What other events do we want to add here?

Let's add new item to the list with a button click. What events can we trigger on the new item? How can we use event delegation to capture events on these new items?

EVENTS & JQUERY

- JavaScript can respond to hundreds of events
- You can get a “handle” on them with jQuery
- The `.on()` method is your go-to method for attaching event handlers
- Delegation can reduce the number of handlers you need to attach
- Delegation can capture events for elements not in the page when the handler is registered
- Events bubble from the in-most outward

JQUERY EFFECTS

JQUERY EFFECTS

- Make your DOM elements come alive!
- jQuery handles the iteration of small changes
- Supports different styles of “easing”
- Provide smoother user interaction
- Create delightful experiences

JQUERY EFFECTS

- How is this useful?
 - Zoom images
 - Transition photos in a slider
 - Animate sorting
 - Build your own customized tooltips

check out these 5+ year old demos of what you can do:

- <http://www.starplugins.com/cloudzoom/examples#>
- <http://aviathemes.com/aviaslider/?slider=winding-blocks>
- <http://razorjack.net/quicksand/>
- <http://demo.tutorialzine.com/2010/04/slideshow-context-tips-jquery-css3/demo.html>

JQUERY EFFECTS

- How is it used?
- Run through it step-by-step.

EFFECTS DEMO

Let's look over the sample code.

Pages: 344 (Form to add a new todo item), 353 (Track position while scrolling)

RECAP

RECAP

- Web apps are powered by async callbacks
- All you need is `.on(...)`
- Have fun with effects!

RESOURCES AND ATTRIBUTION

- <http://cdn.meme.am/instances/59585988.jpg>
- <http://electric-hearts-war.tumblr.com/post/120346550789>
- Effects demos: <http://www.webdesignerdepot.com/2010/07/20-demos-showing-advanced-jquery-effects/>