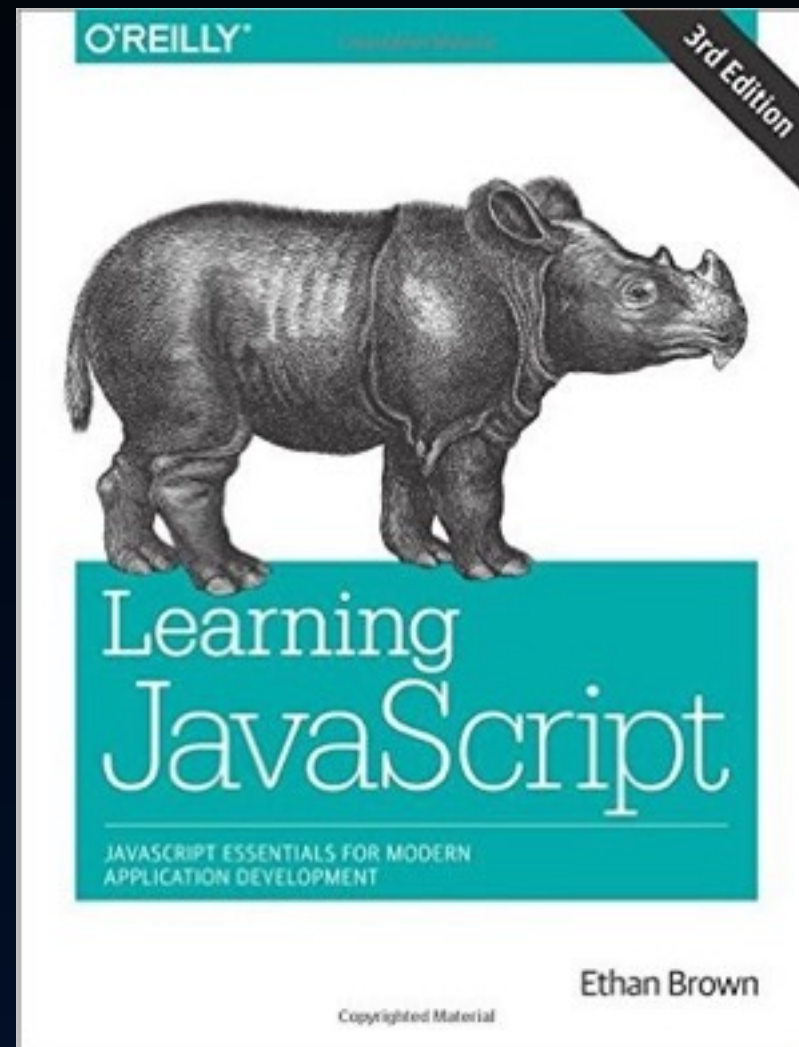# CS50C Web Dev 3

Fall 2017 ~ Ethan Wilde   *Week 4*

SANTA ROSA
JUNIOR COLLEGE

Textbook

Learning JavaScript
Ethan Brown
ISBN 978-1-491-91491-5

Textbook

- **Reading for Week 4**

- ***Learning JavaScript***

  - **Ch 18: JavaScript in the Browser, pp. 255-269**

  - **Ch 19: jQuery**

jQuery Basics

1. **History**

2. **Versions**

3. **Using jQuery to find objects**

4. **jQuery selectors**

5. **What jQuery returns**

6. **Useful built-in jQuery methods**

jQuery Basics

History of jQuery Library

version	date
1	Aug 2006
1.9	Jan 2013
2	Apr 2013
2.2	Jan 2016
3	Jun 2016
3.1	Jul 2016

10+ years

jQuery Basics

Versions

jQuery v1	jQuery v2	jQuery v3
works with older browsers incl. IE 6, 7 + 8 browsers	leaves behind older Internet Explorer 6, 7 + 8	jQuery-Compat 3.0.0 for IE 8, Safari 5, etc.
bigger, slower	smaller, faster	jQuery 3.0.0 for cutting-edge browser versions
n / a	https://blog.jquery.com/2013/04/18/jquery-2-0-released/	https://blog.jquery.com/2014/10/29/jquery-3-0-the-next-generations/

jQuery Basics

1. **History**

2. **Versions**

3. **Using jQuery to find objects**

4. **jQuery selectors**

5. **What jQuery returns**

6. **Useful built-in jQuery methods**

Using jQuery
to Find Objects

jQuery is designed to find objects in HTML pages

Your code asks jQuery to find one or more objects by passing it a _selector_

jQuery Basics

jQuery Selectors

```
$(document).ready();

$('.pickles').html('<p>hey</p>');

$('#thing').clear();
```

DOM OBJECTS OR CSS SELECTORS

jQuery Selectors

```
<p class='pickles'>Happy</p>

<script>
$('.pickles').css('color', 'red');
</script>
```

CSS CLASS SELECTORS

jQuery Selectors

```
<p id='thing'>Happy</p>

<script>
$('#thing').css('color', 'red');
</script>
```

CSS ID SELECTORS

jQuery Basics

1. **History**

2. **Versions**

3. **Using jQuery to find objects**

4. **jQuery selectors**

5. **What jQuery returns**

6. **Useful built-in jQuery methods**

What jQuery Returns

```
<p class='pickles'>Happy</p>
<p class='pickles'>Sad</p>
<p class='pickles'>Mellow</p>


<script>
console.log( $('.pickles') );
</script>
```
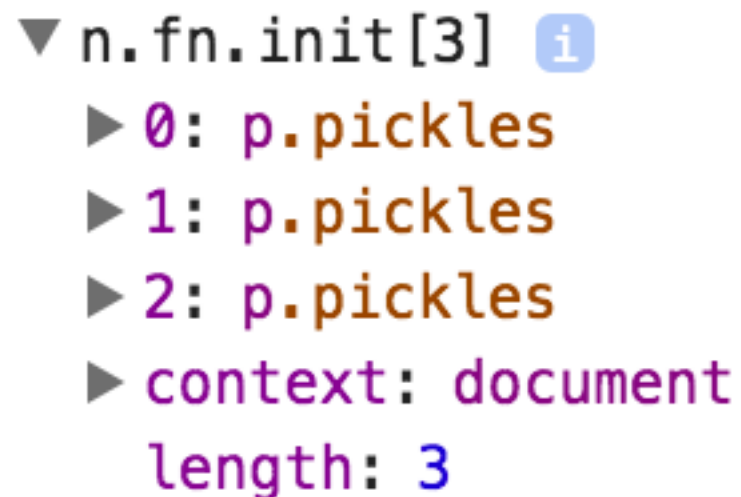
JQUERY RETURNS AN ARRAY OF OBJECTS

What jQuery Returns

```
<script>
console.log( $('.pickles') );
</script>
```



```
▼ n.fn.init[3]  ⓘ
  ▶ 0: p.pickles
  ▶ 1: p.pickles
  ▶ 2: p.pickles
  ▶ context: document
    length: 3
```

JQUERY RETURNS AN ARRAY OF OBJECTS

jQuery Basics

Useful Built-in jQuery Methods *+ Properties*

1. **.length**
   testing for existence of matched objects

2. **.append() vs .html() vs .text()**

3. **.clear()**

4. **.addClass(), .removeClass() and .hasClass()**

5. **.height() and .width()**

6. **.css()**

7. **.animate()**

   PRECEDE EACH WITH *$(selector)*

Useful Built-in jQuery
Events + Loops

1. **.click()**
user events

2. **.ready()**
browser events

3. **.each()**
iterates through object (or other) arrays

4. **.this**
current object target of event
or each() loop

PRECEDE EACH WITH *$(selector)*

JavaScript in Web Browsers

```
document.write( 'Frank' );

window.location.href =
'http://santarosa.edu';
```

DOT NOTATION + THE OBJECT MODEL

JavaScript Coding in Cloud9

```
 4 ▾ const data = [
 5 ▾     {
 6           question:"What is the addition operator?",
 7           answers:[ "-", "+", "/", "*" ],
 8           correct_choice:1
 9      },
10 ▾     {
11           question:"What is the object name for the web page?",
12           answers:[ "window", "location", "document", "navigator"],
13           correct_choice:2
14      }
15   ]
```

Missing semicolon. ..ck of which element in array is current

Cloud9 offers built-in JavaScript debugging of and syntax checking.

Validating JavaScript Code

esprima.org/demo/validate.html

Esprima   Demo ▼   Project ▼   Documentation ▼

Syntax Validator checks for mistakes and errors

```
 1  let x = { world: {
 2      sky: true,
 3      land: true,
 4      hours: 24,
 5      sea : {
 6          color: "blue",
 7          actions: {
 8              wave: function() {
 9                  // do something
10              }
11          },
12          temp: 72.5
13      }
14  }
15  }
16
```

Unlike a typical code linter, this syntax validator does **not** care about coding styles and formatting.

If there is a syntax error, the sign ⚠ will be shown in the left-side gutter. Placing the mouse cursor over that sign will reveal the complete error description.

For a command-line usage, check `esvalidate` from Esprima package (for Node.js). There is also a plugin for Grunt called grunt-jsvalidate. Ant users can take a look at an exemplary Ant task for syntax validation.

Code is syntactically valid.

__esprima.org__ offers live JavaScript code validation.

Modeling and Working with Values

"Imagine a data model for a multiple choice quiz…"

Display an array of object values using jQuery's built-in methods.