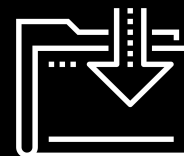




Introduction to ES6

Web Development Boot Camp
Lesson 9.1





There are so many
confusing things about
Javascript

Javascript is not its “official” name

Actually standardized as ECMAScript

- Created in 1995 in 10 days
- JavaScript was originally named JavaScript in hopes of capitalizing on the success of Java.
- Netscape then submitted JavaScript to ECMA International for Standardization. (ECMA is an organization that standardizes information)
- This results in a new language standard, known as **ECMAScript**.



ECMAScript is the ***standard***.
Javascript is an
implementation of that
standard



Change is exciting, but unstable



The **web browser** is the most widespread and easily supported user interface

Javascript is *the* web browser language

Nothing else runs logic in the browser

- The internet is a young platform
- Changing rapidly as we discover new ways to deliver information and entertainment.
- In 2006 JavaScript becomes essential as XMLHttpRequest is standardized.
AJAX became the future of web programming.
- In 2008, the iPhone and mobile computing over limited bandwidth changes the world.

...But Javascript was originally written in 10 days...

ECMAScript standard is rapidly changing

Why? Computing is changing.

- **Cloud computing & mobile clients**
Everything is done over a wire...bandwidth is limited...
- **Software development is now continuous**
Automation and dev-ops
- **Software has to scale almost instantaneously**
Virtual servers and containers
- **Asynchronous, remote procedure calls**
Federated applications, not everything is in one place
- **Moving away from monolithic software to microservices**
Easier to manage changes and load
- **Big software companies contributing to open source Javascript**
Google Chrome changed everything

ECMAScript standard is rapidly changing

Why?

Efficiency.

1. Code needs to be smaller
2. Code needs to be delivered quicker
3. Fewer bugs
 - a. better coding discipline
 - b. more predictable

...and Javascript is not just in the browser anymore.

ECMAScript versions

We are mainly concerned with 3 versions in this class

Note: “ES” is short for “ECMAScript”

1. **ES5** - this is the Javascript we have been learning to this point.
2. **ES6**, aka ES2015 - Big update. We start today.

...[ES7, aka ES2016 was a smallish update]...
3. **ES8**, aka ES2017 - `async/await`, (on Saturday)

Javascript engine support lags

Not all ECMAScript features are supported

V8 is the Javascript engine for Chrome and node.

- It is what turns Javascript from english language looking script into CPU instructions

Chrome V8 and node V8 are different versions

- My current Chrome (stable Mac build) has *8.1.307.31*
- My current node (v12 LTS) has *7.8.279.23-node.34*

How to guarantee compatibility?

Modern ECMAScript is not backwards compatible

We *transpile* modern ECMAScript into ES5
(this is why you need to learn ES5 first)

The Javascript transpiler is called **Babel**.