ECMAScript

- Vishal Avalani

What is ECMAScript?

- The ECMAScript specification is a standardized specification of a scripting language developed by Brendan Eich of Netscape; initially it was named Mocha, later LiveScript, and finally JavaScript.
- In December 1995, Sun Microsystems and Netscape announced JavaScript in a press release.
- **Versions**:
  - 4th Edition
  - 5th Edition
  - 6th Edition - ECMAScript ES6 or ECMAScript 2015
  - 7th Edition - ECMAScript 2016
  - 8th Edition - ECMAScript 2017
  - 9th Edition - ECMAScript 2018
  - 10th Edition - ECMAScript 2019

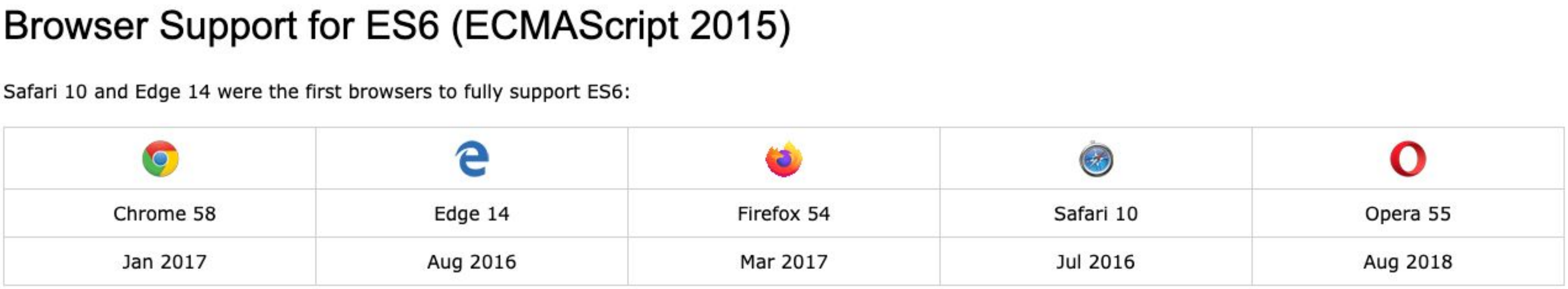
6th Edition - ECMAScript 2015

The 6th edition, initially known as ECMAScript 6 (ES6) then and later renamed to ECMAScript 2015, was finalized in June 2015.

- Class declaration
- Python-style generators, arrow function expression (() => {...})

- let keyword for local declarations, const keyword for constant variable declarations, binary data, typed arrays, new collections (maps, sets and WeakMap), promises, number and math enhancements, reflection, proxies etc.

ES6 Browser Support



Browser Support for ES6 (ECMAScript 2015)

Safari 10 and Edge 14 were the first browsers to fully support ES6:

Chrome 58	Edge 14	Firefox 54	Safari 10	Opera 55
Jan 2017	Aug 2016	Mar 2017	Jul 2016	Aug 2018

https://caniuse.com/

Transpilation

Transpilation is a source-to-source compilation in which the newer versions of JavaScript are used in the user's source code and the transpiler rewrites them so that they are compliant with the current specification.

Usually, transpilers transpile down to **ES3** to maintain compatibility with all versions of browsers. The settings to transpiling to a specific version can be configured according to need. Transpiling adds an extra step to the build process and is sometimes done to avoid needing polyfills.

Polyfills allow using functionalities from newer ECMA versions in older environments that lack them. Polyfills do this at runtime in the interpreter, such as the user's browser or on the server. Instead, transpiling rewrites the ECMA code itself during the build phase of development, before it reaches the interpreter.

7th Edition - ECMAScript 2016

The 7th edition, officially known as ECMAScript 2016, was finalized in June 2016.

The major standard language features include

- block-scoping of variables and functions
- destructuring patterns (of variables)
- proper tail calls
- exponentiation operator ** for numbers
- await, async keywords for asynchronous programming

8th Edition - ECMAScript 2017

The 8th edition, officially known as ECMAScript 2017, was finalized in June 2017.

- Includes async/await constructions, which work using generators and promises.
- ECMAScript 2017 (ES2017), the eighth edition, includes features for concurrency and atomics, syntactic integration with promises (async/await)

9th Edition - ECMAScript 2018

The 9th edition, officially known as ECMAScript 2018, was finalized in June 2018.

- New features include rest/spread operators for variables (three dots: ...identifier)
- asynchronous iteration
- Promise.prototype.finally() and additions to RegExp

10th Edition - ECMAScript 2019

The 10th edition, officially known as ECMAScript 2019, was published in June 2019.

- Array.prototype.flat
- Array.prototype.flatMap
- changes to Array.sort
- Object.fromEntries

Useful topics

1. let/const/var
2. Object oriented Programming
3. String
4. Arrays
5. Destructuring
6. Functions(arrow based expressions)
7. Promises
8. Async/Await
9. Module Import Exports
10. Symbol, Generators, Proxies, Set/Map - Out of scope and not much useful in our training

Reference Links

- https://www.tutorialspoint.com/es6/index.htm
- https://www.w3schools.com/js/js_es6.asp
- https://www.javascripttutorial.net/es6/

Thank You!