

DWA_02.8 Knowledge Check_DWA2

1. What do ES5, ES6 and ES2015 mean - and what are the differences between them?

ES5, ES6, and ES2015 are all references to different versions of the ECMAScript specification, which is the standard for the JavaScript programming language.

ES5 short for ECMAScript 5 released in 2009 which introduced significant updates to JavaScript, including strict mode, which enforces stricter rules for writing JavaScript code, supports for JSON (JavaScript Object Notation), new array manipulation methods like `forEach`, `map`, `filter`, and `reduce`, and improved handling of functions and objects.

ES6 short for ECMAScript 6 released in 2015 and because of this it is also known as ES2015 which brought significant improvements to JavaScript like block-scoped variables with `let` and `const`, arrow functions, template literals for more expressive string interpolation, class syntax for object-oriented programming, modules for organizing and sharing code, destructuring assignments, and more.

2. What are JScript, ActionScript and ECMAScript - and how do they relate to JavaScript?

JScript is a javascript alternative built by **Microsoft** for their compiler "Internet Explorer" because JavaScript was exclusively a proprietary scripting language and safe-guarded internally by the Netscape team.

ActionScript was one of the JavaScript knock-off languages created by **Macromedia** due to the success of *Netscape Navigator* and the popularity of *Javascript*.

ECMAScript is a scripting language specification standardized by ECMA International. The specification is a collection of documents describing how JavaScript and its variants (programming languages like ActionScript and JScript.) should work.

3. What is an example of a JavaScript specification - and where can you find it?

ECMA 262: 1st Edition (1997) and you can find it here: (**ECMAScript: A general purpose, cross-platform programming language**)

4. What are v8, SpiderMonkey, Chakra and Tamarin? Do they run JavaScript differently?

They are all javascript engines or compilers and as they all aim to execute JavaScript code effectively, each engine has its own set of optimizations, memory management techniques, strategies for handling JavaScript features and implementations: "... in some cases, teams either flatly decline to implement aspects of the specification. For example, **Mozilla**, responsible for the Spider Monkey compiler, wrote a **public letter in 2015 on why they refuse to implement some aspects of the latest specification** - specifically citing grievances with the team behind the V8 compiler."

5. Show a practical example using caniuse.com and the MDN compatibility table.

The screenshot shows the CanIUse website interface for the 'alert' feature. The search bar at the top contains 'alert'. Below the search bar, the title 'Window API: alert' is displayed. The 'Usage' section shows 'Global' with a '97.59%' usage percentage. A table below lists various browsers and their compatibility status for the 'alert' feature. The table has columns for Chrome, Edge, Safari, Firefox, Opera, IE, Chrome for Android, Safari on iOS, Samsung Internet, Opera Mini, Opera Mobile, UC Browser for Android, Android Browser, Firefox for Android, QQ Browser, Baidu Browser, and KaiOS Browser. The compatibility status is indicated by colored boxes: green for 'Yes', red for 'No', and grey for 'Partial'. The 'alert' feature is supported in all listed browsers.

Chrome	Edge	Safari	Firefox	Opera	IE	Chrome for Android	Safari on iOS	Samsung Internet	Opera Mini	Opera Mobile	UC Browser for Android	Android Browser	Firefox for Android	QQ Browser	Baidu Browser	KaiOS Browser
4-112	12-112	3.1-16.4	2-112	10-98	6-10		3.2-16.4	4-20		12-12.1		2.1-4.3	4.4-4.4.4			2.5
113	113	16.5	113	99	11	113	16.5	21	all	73	13.4	113	113	13.1	13.18	3.1
114-116		16.6-TP	114-115													

Notes: Test on a real browser Feedback

See full reference on [MDN Web Docs](https://developer.mozilla.org/en-US/docs/Web/API/alert).

Starting with Chrome 46, this method is blocked inside an <iframe> unless its sandbox attribute has the value allow-modals.

Starting with Opera 33, this method is blocked inside an <iframe> unless its sandbox attribute has the value allow-modals.