

DWA_02.8 Knowledge Check_DWA2

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

ES5 is the abbreviation for ECMAScript 5 which was created in December 2009 and it introduced major improvements from the initial ECMAScript 3.

ES6 is the same as ES2015 and this means that this is the sixth ECMAScript edition that was created and published in June 2015. This edition introduced new changes and additions to the language such as the `let` and `const` for variable declaration, arrow functions, destructuring, classes. This edition enhanced the way developers write and structure their code.

The only difference with ES2015 to ES6 is that instead of denoting it with the edition number it uses the year it was published to differentiate it otherwise it is the same as ES6.

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

JavaScript is a type of computer language that follows certain rules called ECMAScript. JScript is a Microsoft version of this language, and ActionScript is another related language, often used for special stuff in Adobe Flash. They're all based on the same set of rules called ECMAScript.

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

An example of a JavaScript specification is the use of the `` tag, this is there to guide developers to add images on the web. Another one is the adding of comments.

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

These are known as JavaScript compilers, and they all run and execute JavaScript differently from the others. However, they all adhere to the ECMAScript standard,

ensuring that JavaScript code runs correctly regardless of the engine. Over time, these engines have evolved to improve performance, memory management, and support for the latest ECMAScript features.

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

Can I use object ? [Settings](#)

50 results found

☒ Caniuse (7) ☒ MDN (43)

Object.values method [OTHER](#)

Usage % of all users Global 96.04%

The `object.values()` method returns an array of a given object's own enumerable property values.

Current aligned Usage relative Date relative Filtered All

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	KalOS Browser
4-53	12-13	3.1-10	2-46	10-40			3.2-10.2	4-5.4								
54-116	14-116	10.1-16.6	47-116	41-101	6-10		10.3-16.6	6.2-21		12-12.1		2.1-4.4.4				2.5
117	117	17.0	117	102	11	117	17.0	22	all	73	15.5	117	117	13.1	13.18	3.1
118-120		17.1-TP	118-120				17.1									

Notes Test on a real browser Known issues (0) Resources (4) Feedback