DWA_02.8 Knowledge Check_DWA2

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

ES stands for ECMAScript and the number being the version. It is a scripting language that is used for client-side scripting of the Web and is defined by ECMA international. The scripting language was created to standardize Javascript as in the 90s there were different scripting languages being used by Microsoft (jScript which ran on Internet explorer) and Netscape who had exclusive proprietary to Javascript back then. ES5 was released in 2009 whereas ES6/ES2015 (used interchangeably) are the same where six is sequential to the previous edition and 2015 being the year that ES6 was released (comes down to how one chooses to refer to the version).

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

Jscript is a scripting language that was developed and used by Microsoft for its Internet explorer. ActionScript, on the other hand, was also a scripting language but it was created by Macromedia which was acquired by adobe and used in developing websites and software for adobe flash. ECMAScript The scripting language was created to standardize scripting languages as in the 90s there were different scripting languages being used by Microsoft (jScript which ran on Internet explorer) and Netscape who had exclusive proprietary to Javascript back then.

They are all scripting languages created by different companies but they were essentially children to JS.

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

An example of a Javascript specification are Tokens and specifically Reserved words which cannot be used as identifiers and they include break, for, else etc. They can be found in ECMA 262: 1st Edition (1997) which are essentially guidelines for ECMAScript.

---

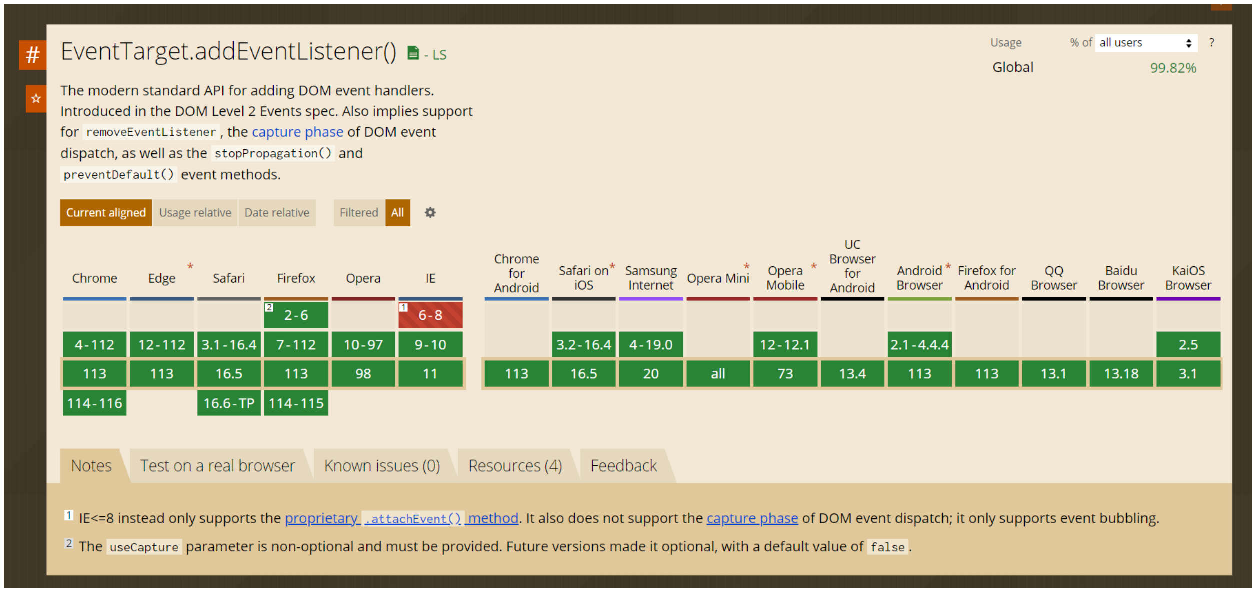
4. What are v8, SpiderMonkey, Chakra and Tamarin? Do they run JavaScript differently?

They are JS compilers that translate JS source code (at run time not before) into machine code and were developed and used by different companies/teams.

---

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

Can I use a feature that you might to check and see how supported it is on different browsers and that allows to to build products and develop code that is widely supported.



EventTarget.addEventListener() - LS

Usage % of all users ?

Global 99.82%

The modern standard API for adding DOM event handlers. Introduced in the DOM Level 2 Events spec. Also implies support for `removeEventListener`, the capture phase of DOM event dispatch, as well as the `stopPropagation()` and `preventDefault()` event methods.

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
			[2] 2-6		[1] 6-8											
4-112	12-112	3.1-16.4	7-112	10-97	9-10		3.2-16.4	4-19.0		12-12.1		2.1-4.4.4				2.5
113	113	16.5	113	98	11	113	16.5	20	all	73	13.4	113	113	13.1	13.18	3.1
114-116		16.6-TP	114-115													

[1] IE<=8 instead only supports the proprietary `.attachEvent()` method. It also does not support the capture phase of DOM event dispatch; it only supports event bubbling.

[2] The `useCapture` parameter is non-optional and must be provided. Future versions made it optional, with a default value of `false`.