DWA_02.8 Knowledge Check_DWA2

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

These are the 5th and 6th standard versions of JavaScript or ECMAScript respectively. ES5/ECMAScript 5 was originally proposed as a simpler standard after the ECMAScript 3 standard version. It was known as ECMAScript 3.1, until the year 2009 when it was completed, signed off and renamed to ECMASCRIPT 5.
ECMAScript 6, a new and improved version of JavaScript after ES5, was released in the year 2015 and later renamed ES2015.

The differences between them are:
In ES5 you can only define variables using the var keyword. But ES6 introduces two new ways to define variables let and const. ES5 has lower performance compared to ES6 with its higher performance. Object manipulation in ES5 is slow while in ES6 object manipulation is less time consuming. ES5 uses function and return keyword to define a function while ES6 has a new feature; Arrow functions that don't require the function keyword to define a function.

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

- JScript is a version or dialect of the original JavaScript before it was standardized by ECMA. JScript was created by Microsoft. Microsoft wanted to keep up with the popularity of JavaScript at the time since the was a considerable difference in user experience since the implementation of JS by NetScape.
- ActionScript is an object-oriented programming language originally developed by Macromedia Inc. ActionScript was a programming language that was based on the early draft for ES4; what should have been the fourth standard version of JavaScript/ECMAScript.
- ECMAScript is the name proposed for JavaScript when it was standardized by the ECMA committee. ECMAScript is the standardized language from the then

JavaScript and Jscript. JavaScript is ECMAScript, but the name ECMAScript never caught on hence we use JavaScript to this day.

---

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

6. ECMAScript Data Types and Values
6.1 ECMAScript Language Types:
An ECMAScript language type corresponds to values that are directly manipulated by an ECMAScript programmer using the ECMAScript language. The ECMAScript language types are Undefined, Null, Boolean, String, Symbol, Number, BigInt, and Object. An ECMAScript language value is a value that is characterized by an ECMAScript language type.

JavaScript specification can be found on the ecma website under ECMAScript.

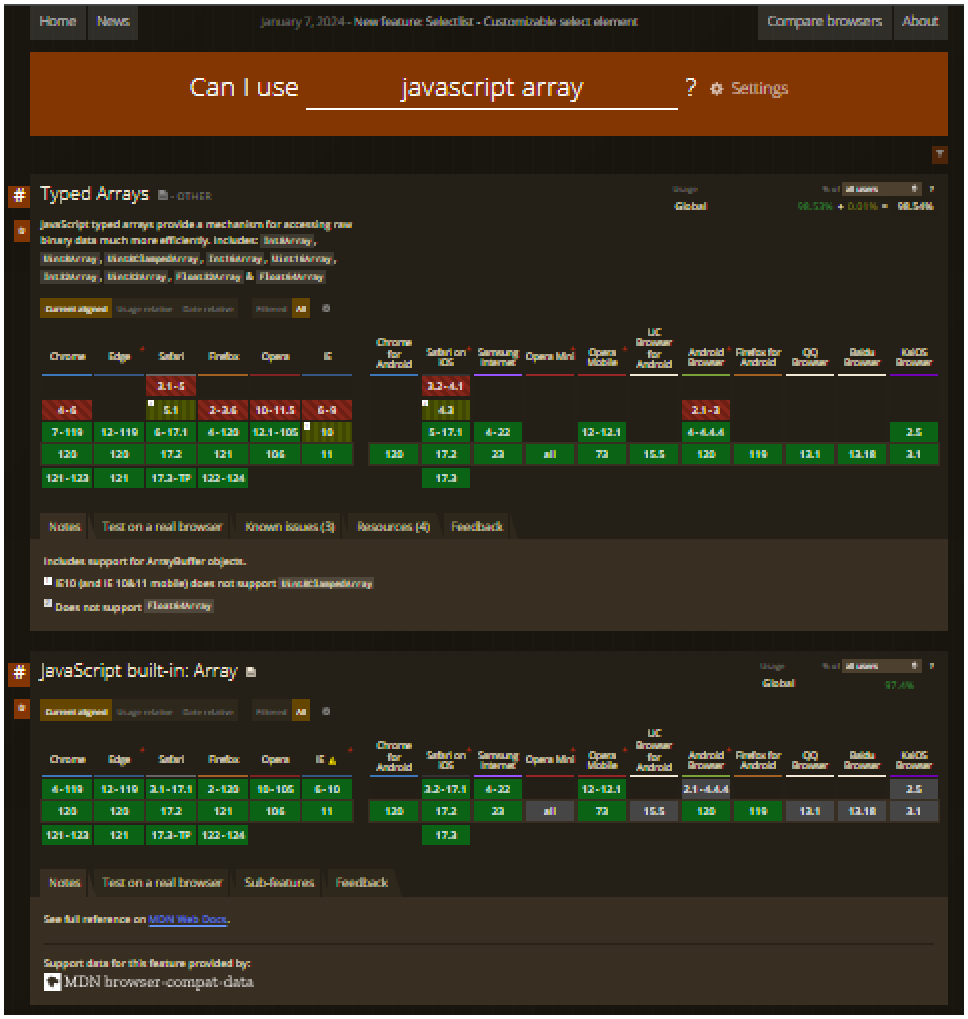
---

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

A JavaScript compiler/ engine is a software component that executes or runs JavaScript code written in a version of the ECMAScript language standard. V8, SpiderMonkey, Chakra and Tamarin are some of the most well-known JavaScript compilers.
They run JavaScript the same way since these examples are Just-in-time (JIT) compilation engines.
JIT is the compilation (translation of a high-level programming language to a low-level programming language to create an executable program) of computer code at runtime (during execution of a program).

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

Caniuse is an open-source project that provides developers with information on browser compatibility. Caniuse can tell you whether a technical feature in your browser is supported by which browsers. Here is an image example of using caniuse and MDN compatibility table to determine how many browsers support JavaScript Array.



January 7, 2024 - New feature: Selectlist - Customizable select element

Can I use javascript array ? Settings

Typed Arrays - OTHER

Usage % of all users
Global 98.53% + 0.01% = 98.54%

JavaScript typed arrays provide a mechanism for accessing raw binary data much more efficiently. Includes: Int8Array, Uint8Array, Uint8ClampedArray, Int16Array, Uint16Array, Int32Array, Uint32Array, Float32Array & Float64Array

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
		3.1 - 5					3.2 - 4.1									
4 - 6		5.1	2 - 3.6	10 - 11.5	6 - 9		4.2					2.1 - 3				
7 - 119	12 - 119	6 - 17.1	4 - 120	12.1 - 105	10		5 - 17.1	4 - 22		12 - 12.1		4 - 4.4.4				2.5
120	120	17.2	121	106	11	120	17.2	23	all	73	15.5	120	119	13.1	13.18	3.1
121 - 123	121	17.3 - TP	122 - 124				17.3									

Includes support for ArrayBuffer objects.

▫ IE10 (and IE 10&11 mobile) does not support Uint8ClampedArray

▫ Does not support Float64Array

JavaScript built-in: Array

Usage % of all users
Global 97.4%

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 - 119	12 - 119	3.1 - 17.1	2 - 120	10 - 105	6 - 10		3.2 - 17.1	4 - 22		12 - 12.1		2.1 - 4.4.4				2.5
120	120	17.2	121	106	11	120	17.2	23	all	73	15.5	120	119	13.1	13.18	3.1
121 - 123	121	17.3 - TP	122 - 124				17.3									

See full reference on MDN Web Docs.

Support data for this feature provided by:
MDN browser-compat-data