DWA_02.8 Knowledge Check_DWA2

---

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

There are three different iterations of the ECMAScript specification for JavaScript: ES5, ES6, and ES2015. While ES6 (or ES2015) provided substantial additions and syntactic changes like arrow functions, classes, and modules, ES5 delivered new features and advancements.

- ES5 (ECMAScript 5) was released in 2009 and introduced several new features and enhancements to JavaScript, such as strict mode, JSON support, and new array methods like `forEach`, `map`, `reduce`, etc.
- ES6 (ECMAScript 2015) is a major update to the language and was released in 2015. It introduced significant additions and syntax improvements, including arrow functions, classes, modules, template literals,destructuring assignments, and more. ES6 is sometimes used interchangeably with ES2015.
- ES2015 is the formal name for the ES6 version of the ECMAScript specification. ECMAScript follows a naming convention where the year of release is used in the specification name.

Subsequent versions were also named in the same pattern, such as ES2016, ES2017, and so on.

---

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

Programming languages related to JavaScript include JScript, ActionScript, and ECMAScript. JavaScript is the most widely used ECMAScript implementation for web development, followed by Microsoft's JScript and ActionScript, which are both based on ECMAScript and used in Adobe Flash.

- JavaScript is Microsoft's implementation of the ECMAScript standard. It was originally developed for Internet Explorer and is similar to JavaScript in terms of syntax and functionality.
- ActionScrIpt is a scripting language developed by Adobe Systems. It is based on ECMAScript and was primarily used for creating interactive content and applications in Adobe Flash.
- ECMAScript is the standard specification that defines the scripting language that JavaScript is based on. JavaScript is the most popular implementation of ECMAScript and is widely used for web development.

---

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

One example of a JavaScript standard is the ECMAScript specification itself. It may be accessed on the ECMA International website and outlines the language's characteristics and norms.

---

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

The JavaScript engines V8, SpiderMonkey, Chakra, and Tamarin are in charge of carrying out JavaScript code. Chrome makes use of V8. Firefox used SpiderMonkey, Edge (which now uses V8) used Chakra, and Flash Player (which is no longer being actively developed) used Tamarin. V8, SpiderMonkey, Chakra, and Tamarin are all different JavaScript engines, which are responsible for executing JavaScript code.

- V8 is the JavaScript engine developed by Google and is used in the Chrome browser and Node.js. It is known for its high performance and is written in C++.

- SpiderMonkey is the JavaScript engine developed by Mozilla and is used in the Firefox browser. It was the first JavaScript engine ever created and is also written in C++.

- Chakra was the JavaScript engine developed by Microsoft for their Edge browser. However, Microsoft has transitioned to using Chromium as the rendering engine for their browser, which includes V8 as the JavaScript engine.

- Tamarin is a JavaScript engine that was developed by Adobe Systems and was used in their Flash Player. However, Flash Player has been deprecated, and Tamarin is no longer actively developed.

While these JavaScript engines may have some differences in their implementation details and performance characteristics, their job is to execute JavaScript code according to the ECMAScript specification.

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

Compatibility tables and the browser compatibility data repository (BCD) MDN has a standard format for tables that illustrate compatibility of shared technologies across all browsers, such as DOM, HTML, CSS, JavaScript, SVG, etc.