DWA_02.8 Knowledge Check_DWA2

---

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

The sixth and fifth editions (ES2015 and ES6 being the same edition under different names) of the ECMA262. ES5 was released in 2009; ES6 was released in 2015. ES5 is the most widely supported edition across browsers and is considered the foundation of JS, added to and expanded upon in subsequent editions.

---

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

JScript (Microsoft) and ActionScript (Macromedia Inc) are reverse engineered forms of the original Javascript created prior to the establishment of ECMA262. ECMA262 is a standard published by ECMA International. It contains the specification (ECMAScript) which dictates what a scripting language could look like, and is what current JavaScript is based on.

---

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

6.1.1 The Undefined Type
The Undefined type has exactly one value, called **undefined**. Any variable that has not been assigned a value has the value **undefined**.

You can browse various editions in the archive at:
https://ecma-international.org/publications-and-standards/standards/ecma-262/

Or see the latest draft of ECMA262 headed by TC39 at: https://tc39.es/ecma262/

---

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

They’re all JavaScript engines for different browsers (though Tamarin was for Adobe Flash Player), developed and maintained by different companies. They differ by the features available to developers, their implementation, and optimisation. Not every feature of the latest version of JavaScript may be supported, and some may have developed their own internal edits that are only available through them. V8 is Google’s JS engine, SpiderMonkey is managed by Firefox and is the original JS engine, and Tamarin was used in Adobe Flash Player (with all our iconic childhood flash games), but ceased development and became obsolete.

---

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

A developer wants to use flexbox, but isn’t sure whether it’s supported across different browsers. By checking caniuse.com you can view a table that shows the level of support of flexbox across different browsers. While this provides information regarding browser support, additional information can be found through the MDN compatibility table which breaks down compatibility even further along with other relevant details.

As of 2019 MDN’s compatibility data is now integrated into caniuse for more convenient access.
