DWA_02.8 Knowledge Check_DWA2

---

1. What do ES5, ES6 and ES2015 mean - and what are the differences between them?
ES5, ES6, and ES2015 refer to different versions of the ECMAScript specification, which is a standard for scripting languages like JavaScript.

ES5 stands for ECMAScript 5, and it was released in 2009. It introduced several new features and enhancements to JavaScript, such as strict mode, JSON support, array methods like forEach, map, and reduce, and improved handling of object properties. ES5 is widely supported by all major browsers.

ES6 stands for ECMAScript 6, and it was released in 2015. It brought significant changes and additions to the JavaScript language. Some of the key features introduced in ES6 include let and const for block-scoped variables, arrow functions, template literals, classes, modules, and destructuring assignments. ES6 also introduced new array and object methods like find, include, and spread syntax.

ES2015 is essentially the same as ES6. ES6 was initially named ES2015 due to the year of its release, but later versions of the ECMAScript specification adopted a yearly naming convention, where each version is named after the year of its release. Therefore, ES6 is often referred to as ES2015, but they both mean the same thing.

The main difference between ES5 and ES6/ES2015 is the set of features and improvements introduced in each version. ES6/ES2015 introduced many new syntactical enhancements and additional functionality to JavaScript, making it easier to write and understand complex code. However, it's important to note that not all ES6/ES2015 features are supported by older browsers, so developers often use transpilers like Babel to convert ES6/ES2015 code into ES5 code that can run in older environments.

---

2. What are JScript, ActionScript and ECMAScript - and how do they relate to JavaScript?
JScript is Microsoft's implementation of the ECMAScript specification. It was developed to be compatible with JavaScript, and it was primarily used in Microsoft's Internet Explorer web browser. JScript was similar to JavaScript but had some differences in terms of language features and browser-specific functionality. With the decline of Internet Explorer, JScript has become less relevant, and modern browsers primarily support JavaScript.

ActionScript is a scripting language developed by Adobe Systems. It is mainly associated with Adobe Flash, which was a popular technology for creating interactive multimedia content on the web. ActionScript is based on ECMAScript, specifically an older version known as ECMAScript 4. ActionScript introduced additional features specific to Flash, such as support for multimedia and vector graphics. However, with the decline of Flash due to security and performance concerns, ActionScript has also become less widely used.

ECMAScript is a scripting language standard that defines the syntax, semantics, and core features of scripting languages like JavaScript. ECMAScript is the official name of the standard, while JavaScript is the most popular implementation of that standard. ECMAScript is maintained by the Ecma International standards organization, and the specification is regularly updated with new features and improvements.

JavaScript is essentially an implementation of the ECMAScript specification, and it is the most widely used scripting language for web development. JavaScript implements the ECMAScript standard and also provides additional features and APIs that are specific to web browsers, allowing developers to interact with web page elements and manipulate the Document Object Model (DOM).

---

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

An example of a JavaScript specification is the ECMAScript specification, which defines the syntax, semantics, and core features of the JavaScript language.

The ECMAScript specification can be found on the Ecma International website, which is the organization responsible for maintaining the specification.

---

4. What are v8, SpiderMonkey, Chakra and Tamarin? Do they run JavaScript differently?
V8, SpiderMonkey, Chakra, and Tamarin are all JavaScript engines, which are responsible for executing JavaScript code in web browsers and other environments. Each engine has its own implementation of the JavaScript language and may have different performance characteristics and features.

V8: V8 is a JavaScript engine developed by Google. It is used in the Chrome web browser and is known for its fast performance. V8 compiles JavaScript code into machine code directly, using techniques like just-in-time (JIT) compilation to optimize performance. V8 also includes features like a garbage collector and supports modern JavaScript language features.

SpiderMonkey: SpiderMonkey is the JavaScript engine developed by Mozilla. It is used in the Firefox web browser. SpiderMonkey was the first JavaScript engine ever created and has been around since the early days of JavaScript. It has undergone significant development and optimization over the years. SpiderMonkey also supports features like JIT compilation and includes various performance optimizations.

Chakra: Chakra is the JavaScript engine initially developed by Microsoft for the Internet Explorer web browser. It has gone through different versions and iterations. Chakra was known for its efficient memory management and was designed to work well with Windows and Microsoft technologies. With the release of Microsoft Edge, the Chakra engine was replaced by a new engine called "EdgeHTML."

Tamarin: Tamarin was a JavaScript engine developed by Adobe Systems. It was specifically designed for the ActionScript and ECMAScript languages. Tamarin included a just-in-time compiler and a garbage collector optimized for efficient memory management. However, Adobe discontinued active development and maintenance of the Tamarin project in 2010.

---

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